

2000

Formal specification of requirements for analytical redundancy-based fault -tolerant flight control systems

Diego Del Gobbo
West Virginia University

Follow this and additional works at: <https://researchrepository.wvu.edu/etd>

Recommended Citation

Del Gobbo, Diego, "Formal specification of requirements for analytical redundancy-based fault -tolerant flight control systems" (2000). *Graduate Theses, Dissertations, and Problem Reports*. 2378.
<https://researchrepository.wvu.edu/etd/2378>

This Dissertation is protected by copyright and/or related rights. It has been brought to you by the The Research Repository @ WVU with permission from the rights-holder(s). You are free to use this Dissertation in any way that is permitted by the copyright and related rights legislation that applies to your use. For other uses you must obtain permission from the rights-holder(s) directly, unless additional rights are indicated by a Creative Commons license in the record and/ or on the work itself. This Dissertation has been accepted for inclusion in WVU Graduate Theses, Dissertations, and Problem Reports collection by an authorized administrator of The Research Repository @ WVU. For more information, please contact researchrepository@mail.wvu.edu.

Formal Specification of Requirements for
Analytical Redundancy based
Fault Tolerant Flight Control Systems

Diego Del Gobbo

Dissertation submitted to the
College of Engineering and Mineral Resources
at West Virginia University
in partial fulfillment of the requirements
for the degree of

Doctor of Philosophy
in
Aerospace Engineering

Marcello Napolitano, Ph.D., Chair
Larry Banta, Ph.D.
Robert Bond, Ph.D.
Ali Mili, Ph.D.
Gary Morris, Ph.D.

Department of Mechanical and Aerospace Engineering

Morgantown, West Virginia
2000

Keywords: Fault Tolerance, Flight Control System, Analytic Redundancy,
System Requirements Specification, Relational Algebra
Copyright 2000 Diego Del Gobbo

Abstract

Formal Specification of Requirements for Analytical Redundancy based Fault Tolerant Flight Control Systems

By Diego Del Gobbo

Flight control systems are undergoing a rapid process of automation. The use of Fly-By-Wire digital flight control systems in commercial aviation (Airbus 320 and Boeing FBW-B777) is a clear sign of this trend. The increased automation goes in parallel with an increased complexity of flight control systems with obvious consequences on reliability and safety. Flight control systems must meet strict fault-tolerance requirements. The standard solution to achieving fault tolerance capability relies on multi-string architectures. On the other hand, multi-string architectures further increase the complexity of the system inducing a reduction of overall reliability.

In the past two decades a variety of techniques based on analytical redundancy have been suggested for fault diagnosis purposes. While research on analytical redundancy has obtained desirable results, a design methodology involving requirements specification and feasibility analysis of analytical redundancy based fault tolerant flight control systems is missing.

The main objective of this research work is to describe within a formal framework the implications of adopting analytical redundancy as a basis to achieve fault tolerance. The research activity involves analysis of the analytical redundancy approach, analysis of flight control system informal requirements, and re-engineering (modeling and specification) of the fault tolerance requirements. The USAF military specification MIL-F-9490D and supporting documents are adopted as source for the flight control informal requirements. The De Havilland DHC-2 general aviation aircraft equipped with standard autopilot control functions is adopted as pilot application. Relational algebra is adopted as formal framework for the specification of the requirements.

The detailed analysis and formalization of the requirements resulted in a better definition of the fault tolerance problem in the framework of analytical redundancy. Fault tolerance requirements and related certification procedures turned out to be considerably more demanding than those typically adopted in the literature. Furthermore, the research work brought up to light important issues in all fields involved in the specification process, namely flight control system requirements, analytical redundancy, and requirements engineering.

Acknowledgments

I would like to thank Dr. Marcello Napolitano, my advisor, for his support during this research. I am thankful to Dr. Ali Mili for having brought some light in the chaos that characterized the early phases of this work. His guidance was crucial to the successful completion of this project. I am also thankful to Dr. Francesco Nasuti for his friendship and for the numerous helpful discussions on the many faces of analytical redundancy.

I wish to thank all of the Drs., researchers, and students who played a role in this research work. Among them I would like to cite Dr. Wu Wen, Dr. Jack Callahan, Dr. Steve Easterbrook, Dr. Bojan Cukic, Dr. Mark Shereshevsky, Dr. Harjinder Sandhu, and Dr. Vittorio Cortellessa. In the years of meetings and discussions since the start of the project, they helped me understand the hidden truth behind a multidisciplinary research work.

I have no words to thank my wife Teresa, without her love and support I would not be here now. Of course, I am grateful to my parents for their unbounded love, and to my grandmother for *"being the origin of the family"*, as she says. I am also grateful to my brothers for not hanging me upside down this time, and to my sister for her unforgettable shout of joy.

Finally, I wish to thank all of those who kept asking: *"So ... have you done?"* ...
I have!

Contents

1	Introduction	1
2	Background information	5
2.1	Issues on the analytical redundancy approach in fault tolerant flight control systems	6
2.1.1	Analytical redundancy	6
2.1.2	Analytical redundancy in flight control systems	9
2.2	Formal specification of system requirements	12
2.2.1	Requirements engineering	12
2.2.2	Advantages of adopting a formal specification language	16
3	Research framework	18
3.1	FTC: the system to be specified	18
3.1.1	FTC environment	19
3.1.2	Main functions of the FTC system	21
3.1.3	FTC interface with its environment	26
3.2	DHC-2 aircraft	29
3.3	Military specification for AFCS	31
4	Formal specification of the FTC environment	35
4.1	Relational specification of elementary requirements	36
4.2	Composition of elementary requirements	45
4.3	Formal specification of the FTC environment	51
4.3.1	Performance requirement composition	54
4.3.2	DHC-2 detail-specification	56
4.3.3	Correctness of AFCS design	60
5	Formal requirements specification of the FTC	62
5.1	FTC requirements	62
5.1.1	FTC functional requirements	62
5.1.2	FTC non-functional requirements	66
5.1.3	FTC-AR requirements	67
5.2	Formal specification of FTC-AR	71
5.2.1	Components partitioning	71
5.2.2	Formal specification of fault hypotheses	74

5.2.3	Relational specification of the FTC-AR requirements	75
5.3	Feasibility analysis	77
5.3.1	Traditional interpretation of detectability and identifiability .	78
5.3.2	Formal definition of detectability and identifiability	80
6	Conclusions	82
A	Predicate Logic and Relational Algebra	92
A.1	Logic	92
A.1.1	Propositional logic	92
A.1.2	Predicate logic	93
A.2	Relational algebra and requirements specification	95
A.2.1	Basics of relational algebra	95
A.2.2	Relational specifications	97
B	Elementary specifications of the AR-FTC environment	102
B.1	Elementary requirements of AFCS performance specification	103
B.2	Elementary requirements of DHC-2 detail-specification	113
B.2.1	DHC-2 airplane dynamics	113
B.2.2	DHC-2 Flight Control System Hardware	119
B.2.3	DHC-2 Flight Control System Software	126
B.3	Fault modes	130
B.3.1	Control-surface fault modes	130
B.3.2	Engine fault modes	131
B.3.3	Actuator fault modes	131
B.3.4	Rate gyro fault modes	132
B.3.5	Accelerometer fault modes	133
B.3.6	Air data sensor fault modes	133
B.3.7	Angle of attack sensor fault modes	134
B.3.8	Attitude and heading sensor fault modes	134
B.4	DHC-2 requirement space restriction sets	136
B.5	Elementary requirements of interface blocks to AR-FTC system . . .	138
C	Support tables of the specification	141

List of Tables


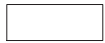






3.1	DHC-2 autopilot functions and related controls	30
4.1	Constants used within the specification of the HH function.	41
4.2	Domain and image variables used within the specification of the HH function.	41
4.3	Quantified variables used within the specification of the HH function.	41
4.4	Predicates and functions used within the specification of the HH function.	41
A.1	Syntax of propositional logic	93
A.2	Semantics of propositional logic	94
A.3	Syntax of predicate logic	94
B.1	Minimum acceptable control accuracy for ALH function	111
C.1	Elementary requirements	143
C.2	Composed requirements	149
C.3	Fault modes	150
C.4	Restriction sets	151
C.5	Spaces used within the requirements specification	152
C.6	Domain and image variables	154
C.7	Constants	161
C.8	Quantified variables	172
C.9	Auxiliary terms	176
C.10	Data-types	182

List of Figures

3.1	Environment of the FTC system.	19
3.2	FTC within its environment.	27
3.3	Block diagram of DHC-2 aircraft and its FCS.	32
4.1	Sample requirements specification structure.	46
4.2	Structure of the AFCS performance requirements specification.	55
4.3	Structure of the DHC-2 detail-specification.	58
5.1	DHC-2 and FTFCS requirements specification structure.	68

List of Symbols and Abbreviations

ACT	Actuator
ADC	Analog to Digital Converter
AFCS	Automatic Flight Control System
ALH	Altitude Hold
AR	Analytical Redundancy
AR-FTFCS	Analytical Redundancy based Fault Tolerant Flight Control System
CP	Control Panel
Cin	Computer input
Cout	Computer output
DAC	Digital to Analog Converter
DHC-2	De Havilland DHC-2 aircraft
DP	Display Panel
FBW	Fly-By-Wire
FCC	Flight Control Computer
FCL	Flight Control Law
FCS	Flight Control System
FCSw	Flight Control Software
FDC	Flight Dynamics and Control
FTC	Fault Tolerance Capability
FTC-ADC	Fault Tolerance Capability – Analog to Digital Converter
FTC-AR	Fault Tolerance Capability – Analytical Redundancy module
FTC-CP	Fault Tolerance Capability – Control Panel module
FTC-DAC	Fault Tolerance Capability – Digital to Analog Converter
FTC-DP	Fault Tolerance Capability – Display Panel module
FTC-IN	Fault Tolerance Capability – software Input interface
FTC-OUT	Fault Tolerance Capability – software Output interface
FTC-SW	Fault Tolerance Capability – Safety switch
FTFCS	Fault Tolerant Flight Control System
HH	Heading Hold
HS	Heading Select
MFCS	Manual Flight Control System
PAH	Pitch Attitude Hold
RAH	Roll Attitude Hold
UAV	Unmanned Aerial Vehicle

	Infallible / fail-operational software component
	Infallible / fail-operational hardware component
	Subsystem made of more than one component
	Fallible hardware component
	Directional data stream
	Infallible / fail-operational software component of FTC system
	Infallible / fail-operational hardware component of FTC system
	Physics law

Note: Symbols used in the document are collected in the tables of Appendix C

Chapter 1

Introduction

The use of Fly-By-Wire (FBW) digital flight control systems is playing a more and more prominent role in commercial aviation. Airbus and Boeing FBW-airliners provide a clear sign of this trend. In FBW technology electronic devices coupled to a digital computer replace conventional mechanical controls. The net result is a more efficient, easier to control aircraft. However, this increased automation goes in parallel with an increased complexity of flight control systems with obvious consequences on reliability and safety. A FBW flight control system is made up of several subsystems including mechanical, electronic, and software components. Each of these subsystems may fail during flight, with disastrous consequences. For this reason flight control systems must meet strict fault-tolerance requirements. The standard solution to achieving fault tolerance capability is the adoption of a multi-string architecture. This architecture is based on redundant units working in parallel and a *voting scheme* that disengages a unit when faulty. Triple and quadruple string architectures are current practice in flight control systems of both military and commercial aviation [62], [21]. On the other hand, multi-string architectures further increase the complexity of the system, induce a reduction of overall reliability, bind to closer maintenance

schedule, and require larger budgets. These factors have induced in recent years an increased interest toward alternative approaches to achieving fault tolerance in flight control systems.

Similar interest comes from fields related to satellite and Unmanned Aerial Vehicle (UAV) applications. Under the ongoing process of globalization the telecommunication industry is growing without rest and commercial satellites are playing an important role in this growth. Weight and size largely affect launching costs of satellites. Weight and size also affect UAV applications. Starting in the late '80s a variety of UAVs have been built for either military or scientific purposes. They vary significantly in size, mission profile, and payload weight carrying capability. With some of them having a payload weight below 20 lbs and dimensions below 15 feet it is clear how weight and room requirements are a major issue. Despite costs, complexity, and weight drawbacks physical redundancy is adopted to achieve fault tolerance.

Redundancy is a must in achieving fault tolerance; the question is whether redundancy other than physical can be adopted. In the past two decades a variety of techniques based on analytical redundancy have been suggested for fault diagnosis purposes. Analytical redundancy identifies with the functional redundancy of the system. No extra hardware is required; fault tolerance is achieved by means of software routines that process sensor outputs and actuator inputs to check for consistency with respect to the analytical model of the system. If an inconsistency is detected, the faulty component is isolated and the control law is reconfigured accordingly. The first analytical redundancy scheme implemented within a flight control systems dates

back to the 70's, when the same aircraft used to conduct research on fly-by-wire technology was also used as testbed for an analytical redundancy management algorithm [56]. The algorithm showed desirable performance during flight test; however, poor robustness to modeling errors and the degree of modeling necessary retrained further development. Since then, a number of results have been obtained in the area of robust fault diagnosis [47]. While research on analytical redundancy has been obtaining desirable results, a design methodology involving requirements specification, feasibility analysis, and certification of analytical redundancy based fault tolerant flight control systems is still missing. Exploring strengths, weaknesses, related degree of reduction of physical redundancy, and overall reliability is a fundamental step in the engineering of such systems.

The main objective of this research is to describe within a formal framework the relevant aspects of Analytical Redundancy based Fault Tolerant Flight Control Systems (AR-FTFCS) to allow the analysis of the implications of adopting the analytical redundancy approach to achieve fault tolerance. The outcome of the research identifies with the requirements specification for an AR-FTFCS.

The De Havilland DHC-2 general aviation airplane equipped with standard autopilot control functions is adopted as pilot application. The steps of the research work are those typical of requirements engineering: analysis of the problem and elicitation of the requirements, requirements modeling, and requirements specification. The USAF military specification MIL-F-9490D [2] and supporting documents are adopted as source for the autopilot performance and fault tolerance requirements.

[37] and [53] are adopted to produce the detail-specification of the DHC-2. The implications of adopting the analytical redundancy approach are analyzed in detail to modify the fault tolerance requirements accordingly. Relational algebra is adopted as formal framework for the specification of the requirements.

Given the multidisciplinary nature of the research work some background information is provided. Analytical redundancy is introduced in Chapter 2; the focus is on the implications of adopting the analytical redundancy approach in flight control systems. The flaws of the *fault-diagnosis* design approach and of the evaluation procedures for AR-FTFCS are highlighted. The second part of the chapter introduces the main concepts of requirements engineering and briefly discusses the advantages of adopting a formal specification language. Appendix A provides a description of predicate logic and relational algebra. The FCS fault tolerance requirements as specified in [2] are illustrated in Chapter 3. In the same chapter the target of the requirements specification is defined and an introductory analysis of the implication of adopting analytical redundancy is performed. Chapter 4 provides a detailed description of the re-engineering and formalization process of the requirements. The composition of the AFCS performance specification and of the DHC-2 detail-specification is illustrated. In Chapter 5 the analysis of the fault tolerance requirements is carried out one step further and the formal specification of the system providing fault tolerance is developed. Appendix B contains the bulk of the specification, while appendix C contains the related supporting tables.

Chapter 2

Background information

This chapter provides some introductory information about two concepts that play a central role in this research work: analytical redundancy and requirements engineering. The first section provides a definition of analytical redundancy and briefly illustrates the most relevant techniques adopting analytical redundancy as a basis for fault tolerance. The focus is mainly on closed loop systems. A discussion about the implications of adopting analytical redundancy to achieve fault tolerance in flight control systems follows. The second section provides an introduction to the requirements engineering discipline. It illustrates the role of requirements specification in the system life-cycle, the main phases of requirements engineering, and the agents involved in the requirements specification process. The chapter closes with a brief discussion about the advantages of adopting a formal specification language.

2.1 Issues on the analytical redundancy approach in fault tolerant flight control systems

2.1.1 Analytical redundancy

Fault tolerance requires some form of redundancy within the system; redundancy provides alternative means to perform a specific task, thus making the system capable of continuing operation despite of localized malfunctions, i.e. of tolerating faults. Two different redundancy approaches are adopted in closed loop system: physical redundancy and analytical redundancy. Physical redundancy is based on a multichannel architecture consisting of three or more intercommunicating systems that are able to work independently. A voting mechanism checks for consistency among the redundant components of each channel. Analytical redundancy identifies with the functional redundancy in the system dynamics. It does not require additional hardware; fault tolerance is achieved by means of software routines that process sensor outputs and actuator inputs to check for consistency with respect to the analytical model of the system. If an inconsistency is detected, the faulty component is isolated and the control law is reconfigured accordingly. Preserved observability allows estimating the measurement of an isolated (allegedly faulty) sensor, while preserved controllability allows controlling the system with an isolated (allegedly faulty) actuator. Numerous survey papers and books [52], [11], and [51] discuss theoretical and practical aspects of adopting the analytical redundancy approach to achieve fault tolerance.

The conceptual structure of an analytical redundancy based fault detection and identification systems consists of two stages: the *residual generation* stage and the

decision making stage [14]. The residuals provide a measure of the inconsistency between the actual behaviour of the system and the system analytical model. Residual values close to zero imply a fault free system; on the other hand, residual values different from zero reveal a fault within the system, and the particular combination of residual values provides means for isolating the faulty component. Processing of the residuals to perform fault detection and isolation is the main task of the decision stage. Decision algorithms range from simple threshold testing on the instantaneous values or on the moving average of the residuals, to more sophisticated statistical testing based on the Generalized Likelihood Ratio test [60], or on the Sequential Probability Ratio test [8]. To achieve fault tolerance an additional recovery stage needs to be added. This stage consists of an adaptive or multi-model control law that processes information provided by the decision making stage to produce a suitable control law. All of the three stages play an equally important role toward the successful fault tolerant control system; however, most of the research focuses on the residual generation problem.

Since the early 70's a variety of residual generation techniques have been suggested in the technical literature. The first techniques adopted a geometric approach that resulted in what is known as the *Beard-Jones Fault Detection Filter* [60]. The detection filters are designed to generate a residual vector with a different direction for each faulty component, thus allowing both detection and isolation. Design issues for such filters are addressed in [46] and [45]. Another approach based on the deterministic description of the system is the *dedicated observer approach* [16]. This

scheme can be adopted for achieving fault tolerance with respect to sensor failures. It is based on a bank of observers each processing a subset of the sensor readings and producing an estimate of the system state vector. Detection and isolation are performed by comparing the state vector estimates produced by the different observers. In its first applications this scheme was implemented by using Luenberger observers; then the scheme was extended to non-linear observers [22], Kalman filters [61], and neural-network based estimators [41]. The *parity relation* approach is based on the design of invariant relations among system inputs and outputs on the basis of the matrices of the system state space model ([15], [12], and [25]). All of the mentioned approaches focus on the system inputs, outputs, or state variables to produce the residual vector. A different approach based on parameter estimation focuses on estimating *unmeasurable* system parameters that are directly related to the source of the fault [33]. The differences among the above techniques are mostly of conceptual nature; studies have shown the practical equivalence of parity relation and observer based approaches [48], and of parity relation and parameter estimation approaches [27].

The weakness of early analytical redundancy based residual generators is in the sensitivity to process disturbances, and in the low performance for non-linear system applications. The *unknown input observer* approach [23] is the first attempt to produce a robust fault detection scheme; it focuses on generating a residual vector that is de-coupled from disturbance inputs. Later on robustness was introduced in the design of parity relation based schemes leading to the *orthogonal parity relation*

concept [28]. Robustness needs lead to a shift of the design into the frequency domain to adopt optimal and robust design techniques like H_∞ [18] and μ synthesis [7]. To address the non-linearity issue researchers extended linear design techniques or adopted approaches based on fuzzy logic [50] and neural networks [57] and [39].

2.1.2 Analytical redundancy in flight control systems

Analytical redundancy approach to fault detection has been adopted in a variety of different fields ranging from automotive engines [26] to electromechanical actuators, induction motor drives, electrical pumps, pipelines [34], chemical processes, heat exchangers [30], gas turbine engines, aircraft jet engine sensors [49], etc. Analytical redundancy has also been used in flight control systems; the very same aircraft used to conduct research on fly-by-wire technology was also used as testbed for an analytical redundancy based fault detection algorithm ([17] and [56]). Since then, a number of results have been published on the suitability of analytical redundancy approach for reconfiguration of flight control systems ([59] and [10]), and for diagnosis of aircraft actuator and sensor failures ([54] and [40]). Nevertheless, doubts remain on the possibility that analytical redundancy based solutions can meet the strict fault tolerance requirements of flight control systems [44]. Section 3.1.2 illustrates such requirements as formulated in the active military specification for piloted flight control systems [2].

There is a considerable difference between the fault tolerance requirements of flight control systems and those of the other applications mentioned above. Fault tolerance in the terms typical to fault detection and identification literature [11] aims at enhancing system reliability and availability by monitoring unreliable components of

the system under the assumption that the other components are working properly. A similar perspective has been erroneously adopted in designing analytical redundancy based fault tolerant in flight control systems. Reliability enhancement and dedicated monitoring of unreliable components are not the key issues in fault tolerant flight control systems. In such systems fault tolerance requires the capability of continued operation after failure of any of the system components (section 6.6 of [3]). Fault-free assumption on any of the system components is not allowed unless failure of these components is proven to be *extremely remote* [2]. Civil and military aircraft equipped with fly-by-wire flight control systems adopt physical redundancy to achieve fault tolerance. Triple and quadruple redundancy is adopted in the Airbus 320 [21] and in the Boeing FBW-B777 [62] to meet fault tolerance requirements. Increased complexity of physical redundant systems brings a degradation of overall system reliability; but the focus is on safety, not on reliability.

Another problem with analytical redundancy is related to the process of performance evaluation of fault tolerant systems adopting such approach. Since these systems exploit the functional redundancy of the plant, when applied in the field of flight control systems they need to be validated over the entire aircraft operational envelope. Instead, most of these solutions are evaluated using a simplified model of the aircraft dynamics, within a limited region of the flight envelope, and with a limited set of maneuvers and fault-modes. Furthermore, evaluation criteria are quite heuristic. A tentative list of criteria for assessing the performance of fault detection and identification systems can be found in [52], and is summarized below:

- promptness of detection
- sensitivity to incipient failure
- false alarm rate
- missed fault detection
- incorrect fault identification

A typical testing procedure for fault detection and identification systems involves injection of a set of failures within a simulation environment and computation of the above indexes. While obtained values can be effectively used to compare the performance of two different solutions, they have no absolute interpretation. The testing environment has a considerable impact on the evaluation of these indexes. Missed detection and false alarm rate do not provide any valuable information if they are not determined within the operational envelope of the system. These figures are highly dependent on the disturbances acting on the system, on the type of fault injected, and – for non linear systems – on the state of the system. Furthermore, the fault could be not detectable at all, thus leading to a missed rate of 100%. But this value is not an index of poor performance of the fault detection system; rather, it indicates a lack of functional redundancy within the system.

In order to provide an objective basis for the evaluation of analytical redundancy based fault tolerant flight control systems it is mandatory to develop the requirements specification for such systems. Validation of a system can be performed only against its specification.

2.2 Formal specification of system requirements

2.2.1 Requirements engineering

A successful system is a system that fully addresses the needs for which it was built. Requirements engineering is the process that discovers those needs, and documents them in a form that is suitable for analysis, communication, and subsequent implementation [42]. For a comprehensive evaluation of the role of the requirements in the development of a system it is important to have a clear understanding of the system life-cycle. The system life-cycle consists of three cycles: the *concept cycle*, the *development cycle*, and the *operation cycle* [35]. The first cycle involves outlining the main functions of the system and investigating its feasibility. The development cycle involves requirements specification, design, implementation, and testing (or certification). The operation cycle spans the time from system certification to retirement from service. The requirements specification phase takes place between the concept phase and the design phase. It transforms the informal, incomplete, and ambiguous needs, as expressed in the concept phase, into a set of requirements that serve as the supporting document for the subsequent phases of design, testing, and operation. The design phase transforms required functions into algorithms and physical processes that are transformed within the implementation phase into software code and hardware components. The testing phase aims at determining whether the system meets all requirements; successful certification implies that the system will meet its operational phase commitments.

Though requirements specification and design play different roles within the de-

velopment cycle they are not separated in time. The system is decomposed into a hierarchy of elements on a functional basis according to the principle of architectural design [19] . Requirements engineering concerns with all elements at all levels. Steps of requirements analysis and design alternate throughout the hierarchical structure of the system to produce a sequence of requirements specifications corresponding to different levels in the decomposition. The prominent role of the requirements specification throughout the development cycle and the consequences of inadequate specification has been widely documented in the literature: *"... No other part of the work so cripples the resulting system if done wrong. No other part is as difficult to rectify later."* [20] *"requirements inadequacies play a major and expensive role in a project failure"* [19].

The main activities of requirements engineering are [19]:

- elicitation
- analysis and modeling
- specification
- validation
- management

Elicitation consists in identifying what problems the system needs to address and in outlining the boundaries between the system and its environment. The modeling activity consists in the development of an abstract description of the system and its

environment. The system environment is the part of the world with which the system will interact and in which the effect of the system are evaluated; its description plays a fundamental role in the requirements specification. System requirements are conditions over phenomena of the environment [36]; *shared phenomena* are the phenomena that belong to both entities while *private phenomena* belong exclusively to the environment. Requirements are conditions over both shared and private quantities. The system can assure satisfaction of requirements involving private quantities thanks to environment properties that relate private phenomena to shared phenomena. These properties are called the *indicative* properties of the environment and they are true irrespective of the presence of the system, as opposed to the *optative* properties that need to be guaranteed by the system and that are captured by the requirements. Only by describing the environment it is possible to describe the purpose of the system and provide the information required to its design.

The specification activity involves the formulation of the requirements by means of a specification language. There is a variety of formal and informal languages adopted in requirements engineering; the advantages of adopting a formal specification language are discussed in the next section. The validation activity consists in establishing whether the requirements specification is complete, correct, unambiguous, consistent, testable, and feasible [9]. Completeness and correctness provide that the requirements specification captures the purpose of the system within its environment. Evaluation of completeness is problematic because it involves a subjective judgment of how well a system that meets the requirement addresses the real-world

need. Absence of ambiguity, consistency, and testability can be obtained by adopting a formal specification language. Requirements feasibility needs to be evaluated by the domain experts. To save efforts and resources it is important to determine ahead of time whether a system can be built that meets the requirements. The last activity of requirements engineering is requirements management. In practice it is impossible to develop a specification that remains stable throughout the life-cycle of the system. Efficient management of requirements plays a crucial role in managing requirements evolution in time and in providing traceability. The *IEEE Guide for Developing System Requirements specifications* [6] provides guidelines to proper structuring of the specification document to facilitate modifications. However, given the considerable dimension that specification documents usually reach, management of specification without well engineered tools is not feasible.

A peculiar feature of requirements engineering is its multidisciplinary nature. Three different agents are typically involved in the requirements engineering process: the customers, the domain experts, and the requirements engineers. The customers are those interested in addressing the real-world problem; they provide a raw definition of the requirements that typically is the result of the concept phase of the system life-cycle. The domain experts are those involved in the activity of design, implementation, integration, and testing. Their contribution in the requirements specification process is essential since they provide the technical know-how to decompose the system into a suitable hierarchical structure, to analyze and model the requirements, and to assess feasibility of the requirements. The requirements engineers work in strict

collaboration with the customers to elicit the requirements and collaborate with the domain experts to perform analysis and modeling of the requirements. Specification, validation, and management are in prevalence tasks of the requirements engineers though some validation tasks involve all three agents. The requirements specification document is the official means of communication between the three agents. It collects all the information required to design the system along with the acceptance criteria that will be used to verify whether the system addresses the real-world need.

2.2.2 Advantages of adopting a formal specification language

The requirements specification involves a considerable amount of engineering analysis and judgement, it is the result of a long sequence of refinements, it is produced with the collaboration of personnel in different area of expertise, and typically results in a voluminous document with a complex set of dependencies. These factors make it difficult to produce a consistent and unambiguous requirements specification. Despite the considerable expressive power of plain-English, its use as specification language introduces an additional source of ambiguity and inconsistency. Considerable leverage can be obtained by adopting a formal specification language. A formal language is a language with a mathematically defined syntax and semantics. The mathematical definition of the language potentially eliminates the ambiguity problem, allows for checking the consistency of the specification, and leads to a specification amenable to automated analysis. Furthermore, the rigid structure of the formal specification serves as a guide in formulating the requirements resulting in a homogeneous document throughout the refinement iterations.

Improvements obtained by adopting a formal language do not come without a cost. Formal specification of a requirement must be explicit in all its parts and this usually results in an even more bulky document. Furthermore, interpretation of a formal specification is not straightforward resulting in a diminished effectiveness of the communication capability of the document.

A suitable specification language should be expressive, that is it should be possible to formalize plain-English requirements without introducing *artifacts*. It should not introduce modeling constraints that could bias the specification structure. It should be monotonic so that the specification can be obtained by composition of sub-specifications; this feature guarantees ease of update during the development cycle. It should be supported by well engineered tools for automatic checking of consistency and for management.

The specification language adopted in this research work is based on predicate logic and relational algebra and is defined in Appendix A

Chapter 3

Research framework

In Chapter 1 it was stated that the target of this research work is the development of a formal requirements specification for a FTFCs where fault tolerance is achieved by exploiting analytical redundancy; in these terms, the target is quite vague and imprecise. In this chapter the target is refined by introducing the environment of the system the author has developed the specification for, and most of all, by elaborating on the function of the system within its environment. In this chapter the author also introduces the aircraft that will serve as pilot application in developing the specification, and the military specification that will serve as main source for AFCS performance and fault tolerance requirements.

3.1 FTC: the system to be specified

In order to specify the requirements for any system it is critical to describe the system environment, mark the boundaries of the system within the environment, and describe the main functions of the system within the environment. The system under analysis is the aggregate of hardware and software components that is added to the AFCS to provide Fault Tolerance Capability (FTC). From now on the acronym FTC is used

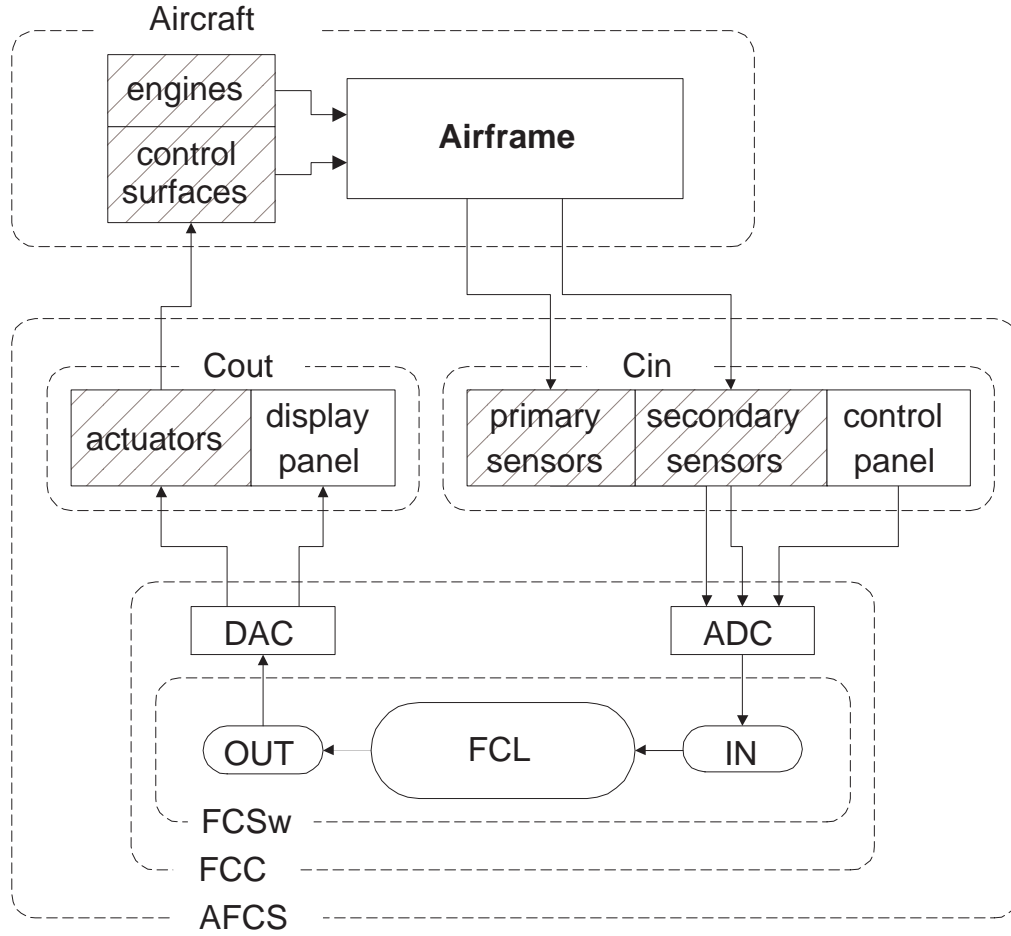


Figure 3.1: Environment of the FTC system.

to identify such system.

3.1.1 FTC environment

The environment of the FTC is the AFCS along with the whole aircraft dynamics. Figure 3.1 shows a functional block diagram of an aircraft equipped with an AFCS. The arrows in the diagram represent data streams like forces and moments, electrical signals, software data, etc; the blocks represent processing units. Square-corner blocks represent hardware units, while round-corner blocks represent software units. Blocks

are grouped by means of dash-lines to form subsystems or systems.

The aircraft system represents the aggregate of airframe, control surfaces, and engines. The control surfaces are typically included in the airframe; here they are separated since different fault hypotheses are introduced for the two units. The airframe block also includes the contribution of gravitational field and air turbulence to the aircraft dynamics. The group of blocks marked AFCS represents the automatic flight control system. It is composed of the flight control computer (FCC), the subsystem processing computer-output (Cout), and the subsystem generating computer-input (Cin). The FCC is composed of the Flight Control Software (FCSw) and of the DAC and ADC blocks. The last two blocks represent the transformation from electrical signals to software data and viceversa. The FCSw is composed of three units: IN, OUT and FCL (Flight Control Law). IN and OUT serve as pre-processing and post-processing units to the flight control law, while FCL is the block that processes software representation of pilot inputs and sensor outputs to produce a software representation of the input to the actuators, the engines, and the display panel. The Cin and Cout subsystems contain blocks whose names are self-explanatory. The set of sensors is separated in *primary* and *secondary* sensors. The primary sensors are those that produce measurements used within the FCL. Measurements from secondary sensors, instead, are used for other purposes, eventually from another control law not shown in the diagram.

The block diagram of Figure 3.1 does not represent the physical units of the AFCS and the related interconnections. Rather, it represents the functions needed within

the AFCS. For example, the FCC block may represent three *physical* computers whose collective behaviour and interface is that of the FCC block.

3.1.2 Main functions of the FTC system

The failure of any component within the FCS can compromise the safety of the aircraft. For this reason FCS's must provide some degree of fault tolerance with respect to failure of their own components. The military specification for FCS's [2] defines three different degrees of fault tolerance that correspond to different degrees of *criticality* of the FCS function. In turn, the criticality of a FCS function is defined in terms of the *operational state* of the aircraft in the post-failure scenario. The relevant operational state – as far as fault tolerance is concerned – is state III, defined as follows:

Operational State III is the state of degraded flight control system performance, safety or reliability which permits safe termination of precision tracking or maneuvering tasks, and safe cruise, descent and landing at the destination of original intent or alternate but where pilot workload is excessive or mission effectiveness is inadequate.

Hence, a FCS function is declared:

Essential if loss of the function results in an unsafe condition or inability to maintain FCS Operational State III

Flight phase essential if loss of the function results in an unsafe condition or inability to maintain FCS Operational State III only during specific flight phases

Non-critical if loss of the function does not affect flight safety or result in control capability below that required for FCS Operational State III

The degrees of fault tolerance for FCS functions are defined as follows:

Fail operational The capability of the FCS for continued operation without degradation following a single failure, and to fail passive in the event of a related subsequent failure.

Fail passive The capability of the FCS to automatically disconnect and to revert to a passive state following a failure.

Fail safe The capability of the FCS in a single channel mode of operation to revert to a safe state following an automatic disconnect in the event of a failure or pilot initiated disconnect.

Each FCS function is required to provide a certain degree of fault tolerance according to its criticality. More specifically, essential FCS functions are required to be fail operational, flight-phase-essential FCS functions are required to be fail passive, and non-critical FCS functions are required to be fail safe. In practice these fault tolerance levels are exceeded for flight-phase-essential and essential controls due to reliability or flight safety requirements. Here the focus is on fault tolerance only and neither reliability nor safety requirements are considered .

The FCS functions under analysis are typical autopilot functions, such as Pitch Attitude Hold, Roll Attitude Hold, etc. Autopilot FCS's are non-critical functions; as such, they are required to be fail safe. Typically, fault tolerance requirements for autopilot functions are met by monitoring values of sensor readings and of control inputs to actuators; if these values are over predetermined ranges the autopilot automatically disengages returning control to the pilot. In this research work autopilot fault tolerance requirements are extended to fail operational capability, with the

constraint of adopting the analytical redundancy approach.

Analytical redundancy based fault tolerance is achieved at the software level; software routines process control law inputs and outputs to check their consistency against an analytical model of the controlled system (in this case the aircraft). Analytical redundancy, however, cannot be used to provide fault tolerance with respect to failure of all the FCS components. Any component of the FCS, either hardware or software, can fail. Analytical redundancy cannot help with software failures; nor it can help if in a single-channel FCS the FCC fails, since the FCC hosts the software that provides fault tolerance. Failure of either the control or the display panel cannot be accommodated at the software level; hence, analytical redundancy is – under these conditions – useless. The remaining components of the FCS are the actuators and the sensors. Analytical redundancy based solutions presented in the literature typically separate the problems of actuator and sensor failure. The rationale behind this choice is simple: fault tolerance with respect to sensor failures is mostly an observation problem, while fault tolerance with respect to actuator failures is mostly a control problem; different expertise and techniques are required for designing the two different FTC systems. The author adopts this modular approach and chooses to focus on sensor failures only. Hence, the FTC system is required to provide fault tolerance with respect to failure of any of the primary sensors. Fault tolerance with respect to failure of the secondary sensors is not required since secondary sensor outputs are not used by the FCL.

Focusing on sensor failures does not imply that all remaining components of the FTC environment are not subject to failure. Whether components are subject to failure or not cannot be arbitrarily established; this is a constraint that is dictated by the nature of the component. Under the realistic assumption that each FCS component is subject to failure the question is which redundancy approach should be adopted to provide fault tolerance. For some of the FCS components fault tolerance cannot be achieved at the software level; these components are the FCSw, the FCC, the CP, and the DP. The author assumes that fault tolerance with respect to failure of these components is achieved by means of physical redundancy, and that performance requirements are still satisfied following a single failure. On the other hand, the author assumes that fault tolerance with respect to actuator and sensor failures is achieved at the software level.

Since analytical redundancy is provided by the functional redundancy within the aircraft dynamics the impact of failure of the aircraft subsystem components must be taken into account as well. Control surfaces and engines are assumed to be fallible, while different hypotheses are made for the airframe. In military aviation partial separation of wing or tail surfaces is possible in a combat scenario; while in commercial aviation this is quite an extraordinary event. For this reason the airframe is assumed infallible. Fallible components whose fault tolerance is not guaranteed by means of physical redundancy are marked by means of oblique lines in figure 3.1 .

In the scenario described above, different modules are used to achieve fault tolerance. These modules adopt either physical or analytical redundancy to provide fault

tolerance with respect to failure of a subset of components of the FCS. It is important to outline to what extent these modules interact with each other. In fact, fault tolerance achieved by means of analytical redundancy is inherently non-modular. The detection, identification, and accommodation tasks are performed by exploiting the correlation among different quantities of the system. This implies that the system relies on the output of other fallible components – eventually monitored by a different FTC module – to achieve fault tolerance with respect to failure of one component . Physical redundancy instead is highly modular. Fault tolerance of each component is achieved by means of similar components that provide means for both fault detection and accommodation. All adopted information is local to the set of redundant components. The dependencies that arise with analytical redundancy make void the modular approach unless the modules are organized in a stratified structure. Within this structure each module builds a new layer of fault tolerant system components. Within the framework of this research one module represents the FTC adopting the physical redundancy approach, while two other modules represent the FTC adopting the analytical redundancy approach to provide fault tolerance with respect to sensor and actuator failures respectively. Physical redundancy is assumed to be exploited first to guarantee correct operation of the hardware hosting and interconnecting to the FCSw. The physical redundancy based FTC module is at the higher level in the FTC stratified structure. Hence, physical redundancy is transparent to the FTC modules that exploit analytical redundancy. The FTC module that provides fault tolerance with respect to sensor failures forms the second layer. This layer produces

validated readings for the monitored sensors. This is the first layer adopting the analytical redundancy approach. For the sake of modularity it must filter out the effects of failures of components that are neither among the monitored sensors, nor among the components whose fault tolerance is provided by the first FTC layer. The last layer is the one providing fault tolerance with respect to actuator failures. This layer can rely on fault tolerant readings from the monitored sensors. However, it must filter out the effects of components that are neither among the actuators, nor among the components whose fault tolerance is provided by the first two FTC layers.

Before illustrating the interface between the FTC and its environment the main functions of the FTC system are summarized as follows:

The FTC system must provide fault tolerance, at the fail-operational degree, with respect to failure of the primary sensors. Fault tolerance must be achieved at the software level, without use of redundant sensors (i.e. exploiting analytical redundancy of the FTC environment). Fault tolerance capability must be achieved regardless of failure of components that are neither among the monitored sensors, nor among those whose fault tolerance is provided by means of physical redundancy. If the FTC is not engaged, the original (without FTC) operation of the AFCS must be guaranteed.

3.1.3 FTC interface with its environment

Figure 3.2 shows how the FTC system fits within its environment. To contain the dimensions of the drawing either acronyms or short-names are used in place of some

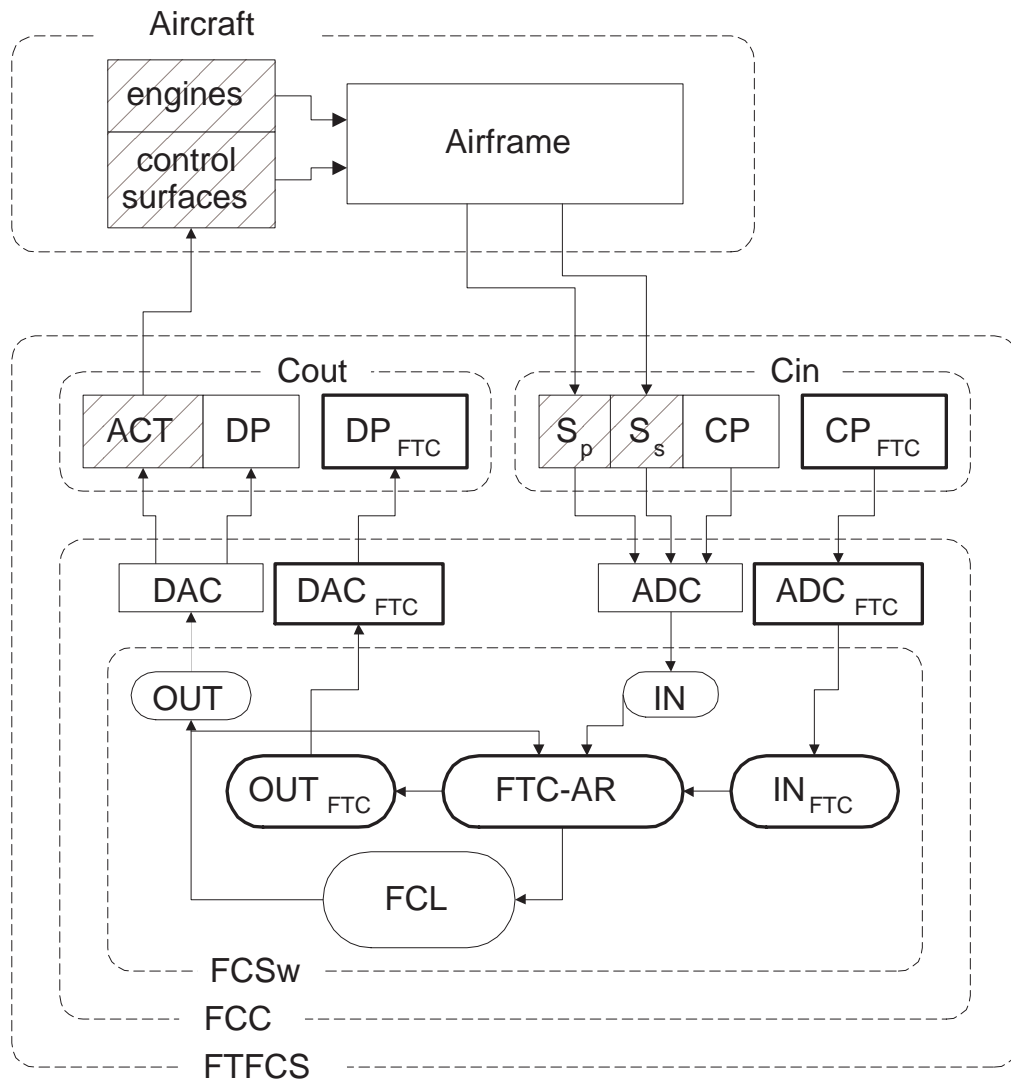


Figure 3.2: FTC within its environment.

of the labels used in figure 3.1. More specifically ACT denotes the actuators, DP and CP denote the display panel and the control panel respectively, and S_p and S_s denote the primary and secondary sensors respectively. The FTC system is composed of the blocks marked with a thicker outline. CP_{FTC} and DP_{FTC} represent the control and display panel of the FTC. They represent the interface with the pilot, and provide means to activate/deactivate the FTC and to signal the operating status (nominal/faulty) of the monitored sensors. ADC_{FTC} and DAC_{FTC} represent the interface between the electrical signals from the FTC control and display panels and the related software variables. IN_{FTC} and OUT_{FTC} represent the software modules that serve as interface between the ADC_{FTC} and DAC_{FTC} blocks, and the FTC-AR block. FTC-AR is the core of the FTC; it represents the software routines that process sensor readings (from the IN block) and control inputs (from the OUT block) to check whether the correlation among them is consistent with the analytical model of the environment. This is the system that exploits analytical redundancy to provide fault tolerance.

With the introduction of the FTC system within the FCS fault tolerance with respect to failure of the FTC components must be guaranteed. These components are of the same kind of those for which physical redundancy was adopted to achieve fault tolerance. Hence, fault tolerance with respect to failure of the FTC components is assumed to be achieved likewise.

3.2 DHC-2 aircraft

In order to develop the requirements specification of a FTC system all relevant details of its environment must be specified. Hence, a pilot application is needed, an aircraft equipped with a FCS that can be adopted as environment for the FTC. The aircraft selected is the De Havilland DHC-2, also known as *Beaver*. This is a general aviation, single engine, high-wing aircraft with a wing span of about 15 meters, fuselage length of about 9 meters, and a maximum take-off weight of about 2300 Kg. Its analytical model, along with its FCS are provided in the Flight Dynamics and Control (FDC) Toolbox for Matlab [38], [37]. Information in [38], [37], and [53] was adopted to provide a description of the blocks of the FTC environment in figure 3.1. The cited documentation provides the analytical model of the DHC-2 aircraft, of the actuator-control-surface chain, the engine, and the continuous-time flight control laws. This information covers the description of the aircraft subsystem, the actuators block, and the FCL block. The description of the environment was completed by developing suitable analytical models for the remaining blocks.

Aerodynamic derivatives and moments of inertia from [37] were adopted for the analytical model of the aircraft; uncertainty bands about nominal values were introduced according to [31]. Actuator-control-surface models include elevators, ailerons, and rudder dynamics; the analytical model of the flaps is not included since the flaps are not used by the autopilot functions. The cited documentation does not contain any sensor model; hence, analytical models were developed from the technical specification of the following sensors:

Table 3.1: DHC-2 autopilot functions and related controls

Autopilot function	Controls
Pitch Attitude Hold (PAH)	SW_{PAH}, θ_r
Altitude Hold (ALH)	SW_{ALH}
Roll Attitude Hold (RAH)	SW_{RAH}, ϕ_r
Heading Hold (HH)	SW_{HH}
Heading Select (HS)	SW_{HS}, ψ_r

Rate gyros and accelerometers *MotionPakTM* Multi-Axis Inertial Sensing System, by BEI, Systron Donner Inertial Division

Angle of attack FAA-authorized commercial airliner angle of attack transducer Series 2568A, by Gulton Statham

Dynamic pressure sensor differential pressure sensor series 142PC05D by Honeywell - Microswitch

Static pressure sensor absolute pressure sensor series 142PC15A, by Honeywell - Microswitch

Attitude and heading sensors FAA authorized Advanced 4MCU IRU, by Honeywell.

The control panel consists of the autopilot control switches and knobs listed in Table 3.1. The Manual Flight Control System (MFCS) controls are omitted since they are not relevant to this study. A generic 16-bit data acquisition card with a ± 10 Volt input and output range was adopted for the ADC and DAC components. Input data to the ADC and output data from the DAC are electrical signals within the ± 10 Volt range. Output data from the ADC and input data to the DAC are the software variables containing the 16-bit counterpart of the related electrical signals. These quantities are named *raw* software variables, as opposed to the *refined* soft-

ware variables representing the value of the measured quantities as used by the flight control laws. The IN and OUT blocks transform raw software variables into refined software variables. Furthermore, the IN block processes pressure and temperature readings to produce air-data (air density, airspeed, barometric altitude) according to the ICAO Standard Atmosphere model. The flight control laws (FCL block) are those implemented within the Flight Dynamics and Control Toolbox for Matlab [37]. The original control laws have been discretized using the forward Euler approximation with a sampling rate $T_s = 1/50s$. The AFCS functions provided with the FDC Matlab Toolbox are listed in Table 3.1. The description of the Flight Control Computer is not provided, it is assumed that the computer provides a suitable environment for hosting the FCSw. Figure 3.3 represents the block diagram of the DHC-2 aircraft and its AFCS. Each component is represented by an identifier composed of the letter 'C' and a subscript that identifies the component.

3.3 Military specification for AFCS

To specify the requirements of the FTC system the military specification MIL-F-9490D [2] is adopted as main source for fault tolerance and performance specification for AFCS. MIL-F-9490D *"Flight Control Systems - Design, Installation and Test of Piloted Aircraft, General Specification for"* [2] is the active specification for FCS for US Air Force manned piloted aircraft. It is supported by other military specifications, standards, handbooks, and non-military publications such as FAA Advisory Circulars, National Aircraft Standards, Technical Reports, etc. The most relevant

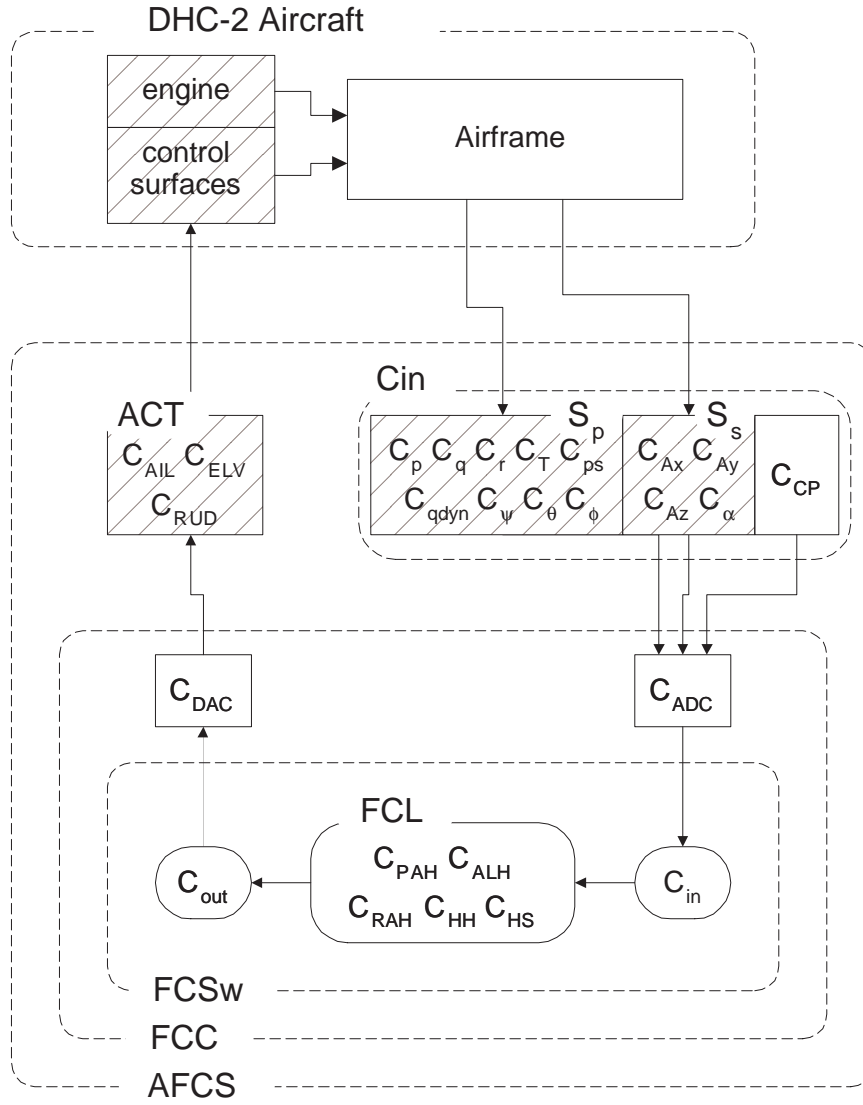


Figure 3.3: Block diagram of DHC-2 aircraft and its FCS.

supporting documents with respect to this research is the military specification MIL-F-8785C *"Flying Qualities of Piloted Airplanes"* [4], the supplement *"Appendix to Background Information and User Guide for MIL-F-9490D"* [3], and the Technical Reports *"Background Information and User guide for MIL-F-9490D"* [1] and *"Background Information and User guide for MIL-F-8785C"* [5].

MIL-F-9490D contains FCS requirements specification (Section 3) along with classification of FCS operational states and of FCS criticality (Section 1), and quality assurance procedures (Section 4). In fact, the document is structured to serve as a guide for all aspects of design, analysis, and test of FCS. The requirements specification spans over the whole system hierarchy, from high-level system requirements (Section 3.1) to subsystem and components requirements (Section 3.2). It covers a wide typology of requirements such as performance requirements for autopilot functions, automatic navigation, ride smoothing, etc.; functional requirements for failure immunity, system test and monitoring, AFCS override, warning and status annunciations, etc.; structural requirements; maintenance requirements; implementation requirements related to technical details such as wiring, shielding, assembling, etc.

For the purpose of developing the requirements specification for the FTC system a narrow subset of requirements has been selected. Appendix B.1 contains the referred military specifications. Selected specifications are reported as they are, with modifications and cuts according to the scope indicated in the sequel. The performance requirements for the following autopilot functions are considered: Pitch Attitude Hold (PAH), Altitude Hold, (ALH), Roll Attitude Hold (RAH), Heading Hold (HH),

and Heading Select (HS). Coordination requirements for lateral-directional control functions, both in steady banked turns and in level flight are considered. Among the functional requirements the focus is on fault-tolerance requirement, limited to those relevant to fail-operational functions. Failure transient requirements are not considered since the focus is on fail-operational capability only.

Specifications contained in [2] do not constitute the whole set of specification for an aircraft; rather, they represent the *aircraft-independent* specifications. *Aircraft-dependent* specifications are collected within the documentation provided by the aircraft manufacturer and are referred to as *detail-specifications*. They specify the operational envelope of the aircraft, the aircraft normal and fault states, the maneuver limits, AFCS functions and their operation such as engagement and disengagement procedures, selection logic, functional safety criteria and limits, and all relevant information related to the specific aircraft. To develop the DHC-2 detail-specification the documentation discussed in the previous section is adopted.

Chapter 4

Formal specification of the FTC environment

The FTC specification is developed on top of the performance specification derived from MIL-F-9490D and of the DHC-2 detail-specification. Since the objective is to develop a formal requirements specification for the FTC, performance and detail specification needs to be formalized first. Relational algebra is adopted as the formal specification framework. It is introduced in Section 4.1; for a more detailed description of relational algebra refer to Appendix A and therein referenced bibliography. To develop the performance and detail formal specification the relevant requirements are decomposed on a functional basis into *elementary requirements*. Each elementary requirement is formalized separately to produce an *elementary specification*. Then, composition operators of relational algebra are used to build up higher level requirements to develop the whole specification. In this chapter the approach toward formalizing the elementary requirements is described in detail. The formalization process is illustrated on one of the elementary requirements from the performance specification. Then, the composition of the elementary specifications is shown using an example. Finally, the requirement structure of the performance and detail specification are

illustrated.

4.1 Relational specification of elementary requirements

Equation 4.1 represents the prototype of a relational specification.

$$\mathcal{R} = \{ (\mathcal{A}, \mathcal{B}) \mid predicate(\mathcal{A}, \mathcal{B}) \} \quad (4.1)$$

This relation represents a set of elements from the space obtained from the cartesian product of the *domain* space \mathcal{A} and the *image* space \mathcal{B} . The space $\mathcal{A} \times \mathcal{B}$ is denoted the *signature* of the relation, and $(\mathcal{A}, \mathcal{B})$ represents the generic element of this space. Boldface font is adopted for the spaces to distinguish them from their respective elements. Each element of a space represents a structure of variables. \mathcal{A} and \mathcal{B} denote the domain and image elements respectively, and their variables the domain and image variables. $predicate(\mathcal{A}, \mathcal{B})$ is a predicate in terms of the domain and image variables, of constants, and – eventually – of quantified variables introduced within the predicate itself. The predicate evaluates either *true* or *false* depending on the value of the domain and image variables. The relation represents the set of couple of elements $(\mathcal{A}, \mathcal{B})$ that cause the predicate to evaluate true.

A requirement can be interpreted as a relationship among some relevant quantities. Consider the space whose coordinates identify with the quantities adopted in the requirement. Each point of this space represents a particular choice of values for those quantities. The required relationship identifies with a region in the space of requirement's quantities, much like a function $y = f(x)$ identifies with a curve in

the space $X \times Y$. This region contains all the elements whose quantities' values satisfy the required relationship. A relation of the type described in equation 4.1 can identify that region – hence specify the related requirement – provided that the adopted constants and variables represent the relevant quantities, and the predicate captures the required relationship among those quantities.

In the requirement formalization process the quantities that are explicitly or implicitly used to formulate the requirement are identified first. Hence, constants and variables to represent those quantities are introduced. Constants are used to represent fixed quantities; image variables are used to represent quantities whose value is somehow constrained by the requirement; domain variables are used to represent quantities whose value delimits the scope of the requirement; quantified variables are used to represent quantities that play a role in the formulation of the requirement but that are neither constrained, nor used to specify the requirements scope. For the purpose of making the relation more readable and the whole specification less repetitive and cumbersome *auxiliary terms* are introduced. These terms represent expressions and functions that are repetitively used within the specification. Finally, a predicate that captures the semantics of the requirement is formulated. This requires the predicate to evaluate true if and only if the required relationship among the requirement quantities holds.

Typically, variables used in relations represent the instantaneous value of the related quantities. This approach allows for specifying requirements in terms of instantaneous input/output relationships, but it is not suitable for specifying AFCS

requirements. Most of AFCS performance requirements are formulated as a constraint over a quantity's time evolution within a certain time interval, rather than over the quantity's instantaneous values. Damping requirements, and RMS-deviation requirements are among two examples. The damping requirement is typically expressed in terms of the damping factor of the *equivalent second order system*. To verify this requirement an identification procedure is used to process system input and output over the relevant time interval and to produce the equivalent system. It is not possible to formulate the damping requirement on the basis of instantaneous input/output values. The RMS-deviation requirement is by definition a constraint over the integral of the output within the relevant time interval. Once again, this requirement cannot be formulated as a constraint over instantaneous values of system output.

To solve this problem the author adopts variables that represent the whole time evolution of the related quantity, rather than its instantaneous values. For example, the variable $\phi()$ is used to represent the time evolution of the bank angle within the time interval $[0, \infty)$. The *empty brackets* $()$ are adopted to indicate that the variable represents the whole time evolution of a quantity rather than the value of the quantity at a specific time instant.

To illustrate the requirements formalization process and provide a guide to interpreting the relational specification the formalization of the Heading Hold (HH) control function requirement is commented. This requirement is fairly simple, yet provides a number of meaningful points for discussion. The plain-English specification from [2]

is reported below; to facilitate the analysis each section of the requirement has been labeled with a letter.

3.1.2.2 Heading Hold

- a) In smooth air, heading shall be maintained within a static accuracy of ± 0.5 degree with respect to the reference.
- b) In turbulence, RMS deviations shall not exceed 5 degrees in heading at the intensities specified in 3.1.3.7.
- c) When heading hold is engaged, the aircraft shall roll towards wings level.
- d) The reference heading shall be that heading that exists when the aircraft passes through a roll attitude that is wings level plus or minus a tolerance.

As first step the plain-English requirement is analyzed to identify the quantities that are used either explicitly or implicitly within the specification. During this analysis the author introduces – sometimes between brackets – the identifiers that will represent those quantities within the relation. The HH requirement specifies accuracy requirements for the heading angle ($\psi()$) for operation in both smooth air and turbulence when the HH function is engaged. Hence, the two operation cases must be separated. To this purpose the auxiliary function $turb(t_a, t_b)$ is introduced. This predicate is expressed in terms of the random and discrete turbulence components of the wind velocity vector $u_{wt}()$, $v_{wt}()$, $w_{wt}()$, $u_{wg}()$, $v_{wg}()$, $w_{wg}()$, and of the time instants t_a and t_b . It returns false if and only if the airplane is operating in smooth air within the time interval $[t_a, t_b]$. Accuracy requirement for operation in turbulence are expressed in terms of the RMS deviation from the reference heading. The auxiliary

function $RMS(dev(), t_a, t_b)$ is introduced; it returns the RMS value of the variable $dev()$ within the specified time interval $[t_a, t_b]$. The relevant time instants of the specification are three. The first one (t_1) is the HH function engagement instant. The second one (t_2) is the time instant when the reference heading (ψ_r) is determined. In fact, the specification requires the airplane to roll ($\phi()$) towards wings level, and fixes the reference heading as *"that heading that exists when the aircraft passes through a roll attitude that is wings level plus or minus a tolerance"* (ϕ_{acc}). The third time instant (t_3) can be any time instant preceding HH function disengagement. Other quantities that need to be represented are the required accuracy levels in both smooth air (ψ_{acc}) and turbulence (ψ_{RMS}) operation, and the HH control switch ($SW_{HH}()$). Another auxiliary function is introduced: $engaged(SW(), t_a, t_b)$; it returns true if control $SW()$ is engaged at $t = t_a$ and stays engaged within the whole time interval $[t_a, t_b]$.

After identifying the relevant quantities the related identifiers are separated into classes of constants, domain, image, and quantified variables, and auxiliary functions. Tables 4.1 through 4.4 list all identifiers used within the HH relational specification.

The signature and the predicate of the HH relational requirement are given by the following two equations:

$$HH_{sign} = ((SW_{HH}(), u_{wt}(), v_{wt}(), w_{wt}()), (\psi(), \phi())) \quad (4.2)$$

Table 4.1: Constants used within the specification of the HH function.

ID and value	Type	Definition
$\psi_{acc} = 0.5 \cdot degree2SI$	angle-T	heading accuracy in smooth air
$\psi_{RMS} = 5 \cdot degree2SI$	angle-T	RMS heading accuracy in turbulence
$\phi_{acc} = 1.0 \cdot degree2SI$	angle-T	roll accuracy in smooth air

Table 4.2: Domain and image variables used within the specification of the HH function.

ID	Type	Definition
$SW_{HH}()$	time-T \rightarrow switch-T	HH autopilot on/off switch
$u_{wt}(), v_{wt}(), w_{wt}()$	time-T \rightarrow velocity-T	wind-turbulence components of wind velocity along body-axes
$u_{wg}(), v_{wg}(), w_{wg}()$	time-T \rightarrow velocity-T	wind-gust components of wind velocity along body-axes
$\psi()$	time-T \rightarrow angle-T	heading angle
$\phi()$	time-T \rightarrow angle-T	bank angle

Table 4.3: Quantified variables used within the specification of the HH function.

ID	Type	Definition
t	time-T	generic time instant
t_1	time-T	HH function engagement time instant
t_2	time-T	time instant when the reference heading is determined
t_3	time-T	time instant delimiting the scope of the requirement
ψ_r	angle-T	reference heading

Table 4.4: Predicates and functions used within the specification of the HH function.

ID	Description
$engaged(SW(), t_a, t_b)$	predicate that evaluates true only if the switch $SW()$ is engaged at $t = t_a$ and stays engaged throughout the time interval $[t_a, t_b]$
$RMS(f(), t_a, t_b)$	Root Mean Square value of the function $f()$ over the time interval $[t_a, t_b]$
$turb(t_a, t_b)$	predicate that evaluates true if random and discrete turbulence wind components are not zero

$$\begin{aligned}
HH_{pred} = & \tag{4.3} \\
& \forall t_1 \forall t_3 \left(\right. \\
& \quad 0 \leq t_1 < t_3 < \infty \wedge \\
& \quad engaged(SW_{HH}(), t_1, t_3) \Rightarrow \\
& \quad \exists t_2 \exists \psi_r \left(\right. \\
& \quad \quad t_1 \leq t_2 < t_3 \wedge \\
& \quad \quad \forall t \left(t_2 \leq t \leq t_3 \Rightarrow |\phi(t)| < \phi_{acc} \right) \wedge \\
& \quad \quad \psi_r = \psi(t_2) \wedge \\
& \quad \quad \neg turb(t_1, t_3) \Rightarrow \forall t \left(t_2 \leq t \leq t_3 \Rightarrow |\psi(t) - \psi_r| < \psi_{acc} \right) \wedge \\
& \quad \quad turb(t_1, t_3) \Rightarrow RMS(\psi() - \psi_r, t_2, t_3) < \psi_{RMS} \\
& \quad \left. \right) \\
& \left. \right)
\end{aligned}$$

Hence, the relational requirement for the HH control function is:

$$\mathcal{R}_{HH} = \left\{ HH_{sign} \mid HH_{pred} \right\} \tag{4.4}$$

The HH predicate 4.3 can be read as follows:

FOR EVERY couple of time instants t_1, t_3

IF

$[t_1, t_3]$ is a time interval within $[0, \infty)$ AND

the HH control function is engaged at $t = t_1$ and stays engaged throughout the interval $[t_1, t_3]$

THEN

there must EXIST a time instant t_2 and a reference heading ψ_r such that

- c_1) t_2 is within the interval $[t_1, t_3)$ AND
- c_2) the bank angle is approximately zero within the time interval $[t_2, t_3)$ AND
- d) the reference heading ψ_r is the heading angle at $t = t_2$ AND
- a) IF there is no turbulence within the time interval $[t_1, t_3]$ THEN the deviation of the heading angle ψ from the referenced heading ψ_r shall not be larger than ψ_{acc} at all time instants within the time interval $[t_2, t_3]$ AND
- b) IF there is turbulence within the time interval $[t_1, t_3]$ THEN the RMS value of the deviation of the heading angle ψ from the referenced heading ψ_r over the time interval $[t_2, t_3]$ shall not be larger than ψ_{RMS}

The labels in the *THEN* section recall the labels inserted in the plain-English specification. Sections a) and b) refer to the accuracy requirement in smooth and turbulence operation respectively. Both sections c_1) and c_2) refer to section c) of the plain-English specification. This section reads "*When heading hold is engaged the aircraft shall roll towards wings level*". This section has been formalized by requiring the airplane to reach – at a certain time instant t_2 – a state with a bank angle *close* to zero. To quantify *how close*, the author adopted the accuracy threshold ϕ_{acc} used in the Roll Attitude Hold (RAH) specification (Section 3.1.2.1 of [2], see also Appendix B.1). This solution is the author's best guess to make up for the lack of a threshold in the plain-English specification. The same solution was adopted to make up for the lack of indications on how to determine the reference heading. In fact, section d)

of the original specification reads *"The reference heading shall be that heading that exists when the aircraft passes through a roll attitude that is wings level plus or minus a tolerance"*. The tolerance is left unspecified. Another problem within the plain-English specification is the lack of a *settling-time* requirement for the airplane to reach wings level. Unfortunately, the settling-time requirement cannot be extracted from the RAH specification since in that specification the settling-time is specified only for the case of a 5 degree attitude disturbance.

While analyzing the AFCS requirements from [2] the author found out that most of them are incomplete and not properly specified. The qualitative nature of the AFCS specification is explicitly stated in Section 3.1.11 of [4]. The qualitative nature of some requirements like *"Entry and exit from the turn shall be smooth and rapid"* (Section 3.1.2.3 in [2]), *"For engagement at rates above 2000 fpm the AFCS shall not cause any unsafe maneuvers"* (Section 3.1.2.5 in [2]), or *"No out of trim condition shall exist at disengagement which cannot be easily controlled by the pilot"* (Section 3.1.3.3.2 in [2]) did not allow to develop a formal specification for the whole plain-English requirement. However, this does not represent a problem for the development of the FTC formal specification. Because of the monotonic nature of relational requirements, the FTC specification can be developed on top of temporary FCS specification, postponing completion of the FCS specification to a later refinement step.

4.2 Composition of elementary requirements

In order to describe the composition of the elementary specifications the composition operators of relational algebra are now introduced along with the concept of *refinement* ordering between requirements and of system *correctness*. Relation \mathcal{R}_1 refines relation \mathcal{R}_2 , denoted by $\mathcal{R}_1 \sqsupseteq \mathcal{R}_2$, if \mathcal{R}_1 specifies a stronger requirement than \mathcal{R}_2 , in the sense that it imposes a constraint over a wider domain and/or it is more specific. System correctness is defined as a refinement constraint between the relation \mathcal{P} representing the system implementation and the relation \mathcal{R} representing the system requirements: \mathcal{P} is correct with respect to \mathcal{R} if $\mathcal{P} \sqsupseteq \mathcal{R}$. The adopted composition operators are the *join* (\sqcup) and the *product* (\circ) operators. The join of two relations represents the *sum* of the requirements. A system \mathcal{P} that meets the requirement specified by $\mathcal{R}_1 \sqcup \mathcal{R}_2$ also meets the requirements specified by \mathcal{R}_1 and \mathcal{R}_2 separately. This is formally stated by the following formula:

$$\mathcal{P} \sqsupseteq \mathcal{R}_1 \sqcup \mathcal{R}_2 \Rightarrow \mathcal{P} \sqsupseteq \mathcal{R}_1 \wedge \mathcal{P} \sqsupseteq \mathcal{R}_2 \quad (4.5)$$

The join operator is adopted to compose relations that capture *parallel* requirements within the specification structure. The product of two relations \mathcal{R}_1 and \mathcal{R}_2 represents the requirement of the sequence of two systems \mathcal{P}_1 and \mathcal{P}_2 whose requirements are specified by \mathcal{R}_1 and \mathcal{R}_2 respectively. If \mathcal{P}_{12} denotes the series of systems \mathcal{P}_1 and \mathcal{P}_2 the main property of the product operator is defined by the following formula:

$$\mathcal{P}_1 \sqsupseteq \mathcal{R}_1 \wedge \mathcal{P}_2 \sqsupseteq \mathcal{R}_2 \Leftrightarrow \mathcal{P}_{12} \sqsupseteq \mathcal{R}_1 \circ \mathcal{R}_2 \quad (4.6)$$

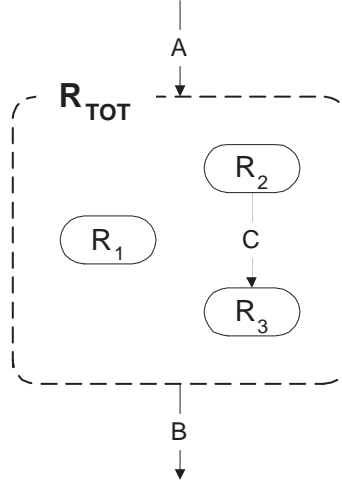


Figure 4.1: Sample requirements specification structure.

The product operator is adopted to compose relations that capture *sequential* requirements within the specification structure.

Figure 4.1 represents a sample specification structure. \mathcal{R}_1 , \mathcal{R}_2 , and \mathcal{R}_3 are the relations that specify the elementary requirements. The decomposition of the whole system into subsystems and components according to a convenient partitioning criterion is what drives the decomposition of the specification into elementary requirements. The same partitioning criterion also determines the nature of the relationship among the specification components. In figure 4.1 relations \mathcal{R}_2 and \mathcal{R}_3 represent sequential requirements; this is graphically denoted by the arrow that directly connects the two relations. \mathcal{R}_1 and the sequence of \mathcal{R}_2 and \mathcal{R}_3 represent parallel requirements. The join of these relations forms a higher level requirement named \mathcal{R}_{TOT} ; this is denoted by the dash-line box that groups the three relations. The other two arrows represent the signature of \mathcal{R}_{TOT} . More precisely, \mathcal{A} represents the domain space, while \mathcal{B} represents the image space. Hence, the whole specification

can be formulated as follows:

$$\mathcal{R}_1 \subseteq \mathcal{A} \times \mathcal{B} \quad (4.7)$$

$$\mathcal{R}_2 \subseteq \mathcal{A} \times \mathcal{C} \quad (4.8)$$

$$\mathcal{R}_3 \subseteq \mathcal{C} \times \mathcal{B} \quad (4.9)$$

$$\mathcal{R}_{TOT} = \mathcal{R}_1 \sqcup \mathcal{R}_2 \circ \mathcal{R}_3 \quad (4.10)$$

Composition operators cannot be applied to any couple of relations. The join operator can be applied only to relations with the same signature. On the other hand, the product operator can be applied only if the image space of the left relational operand coincides with the domain space of the right relational operand. Hence, before composing the relations to form higher level requirements the domain and/or the image spaces of the relations used in the composition might need to be expanded. Since, the expansion operation preserves the property that is captured by the relation, the expanded relation specifies the same requirement as the original relation. A possible approach is to expand the spaces of all relations to the space of all variables that appear in the specification. The expanded relations would form a set of homogeneous relations and both join and product operators could be applied to any couple of expanded relations. However, this approach would lead to a flat set of relations, where the structure of the system is lost.

The adopted expansion approach preserves the structure of the system, with the result of a more informative specification that potentially leads to a more effective validation. To illustrate the expansion process the composition of the Heading Hold (Section 3.1.2.2 in [2]) and the Heading Select (HS) (Section 3.1.2.3 in [2]) requirements

is illustrated. For the expansion operation only the two relation signatures are needed. The signature of the HH relation is given in equation 4.2. The signature of the HS relation from the plain-English requirement is now determined.

3.1.2.3 Heading Select

- a) The aircraft shall automatically turn through the smallest angle
- b) to any heading selected or preselected by the pilot and
- c) maintain that heading to the tolerances specified for heading hold.
- d) The contractor shall determine a bank angle limit which provides a satisfactory turn rate and precludes impending stall.
- e) The aircraft shall not overshoot the selected heading by more than 1.5 degrees.
- f) Entry into and exit from the turn shall be smooth and rapid.
- g) The roll rate shall not exceed 10 deg/sec and
- h) roll acceleration shall not exceed 5 deg/sec/sec.

The requirement imposes a constrain over the roll angle $\phi()$ (section a), over the heading angle $psi()$ (sections c and e), and over the roll rate $p()$ (sections g and h). The scope of the requirement is determined by the HS engagement control switch $SW_{HS}()$ (implicitly assumed activated), by the selected reference heading $\psi_r()$ (section b), and by the wind velocity components $u_{wt}()$, $v_{wt}()$, $w_{wt}()$ (section c) through the reference to the HH requirement. Hence, the signature for the HS relational requirement is:

$$HS_{sign} = \left((SW_{HS}(), \psi_r(), u_{wt}(), v_{wt}(), w_{wt}()), (\psi(), \phi(), p()) \right) \quad (4.11)$$

The signature of the HH relational requirement is reported below:

$$HH_{sign} = \left((SW_{HH}(), u_{wt}(), v_{wt}(), w_{wt}()), (\psi(), \phi()) \right) \quad (4.12)$$

The HH and HS control function requirements are very *close* in the AFCS performance specification structure (see Section 4.3.1). The join of the two relational requirements form the *Heading Control Function* requirement. Nevertheless, the two signatures 4.12 and 4.11 are different. In order to expand the domain and image space of the HH and HS relations while preserving the structure of the specification the author proceeds with two expansion operations. The first expansion is based on partitioning the set of all variables used in the specification into disjoint subsets. This partitioning is performed on the basis of the functional equivalence criterion that drove the decomposition of the systems into subsystems and components. Hence, each occurrence of equivalent variables in the signature is substituted with the corresponding class of equivalence. The classes of equivalence adopted in the specification are listed in Table C.5 along with their respective variables. Among these classes there is the class of pilot input variables \mathcal{U}_p , the class of aircraft state variables \mathcal{X} , and the class of the components of the wind velocity vector \mathcal{U}_w . With the three equivalent classes just introduced the HH and HS relation signatures can be formulated as follows:

$$HH_{sign2} = \left((\mathcal{U}_w, \mathcal{U}_p), \mathcal{X} \right) \quad (4.13)$$

$$HS_{sign2} = \left((\mathcal{U}_w, \mathcal{U}_p), \mathcal{X} \right) \quad (4.14)$$

The two signatures 4.13 and 4.14 are now the same and the two relations can be joined.

For some relations a second expansion operation is needed. This is the case of the sensor requirement \mathcal{R}_{Sp} and of the control panel requirement \mathcal{R}_{CP} in the DHC-2 detail-specification. The two requirements need to be joined as part of the computer input requirement \mathcal{R}_{CIN} . However, they have the following different signatures after the first expansion:

$$\mathcal{R}_{Sign2} = (\mathcal{X}, \tilde{\mathcal{X}}) \quad (4.15)$$

$$\mathcal{R}_{CPsign2} = (\mathcal{U}_p, \tilde{\mathcal{U}}_p) \quad (4.16)$$

$\tilde{\mathcal{X}}$ and $\tilde{\mathcal{U}}_p$ represent the class of electrical signals correlated to aircraft states and pilot inputs respectively. The problem arises because the equivalence partitioning introduced in the first expansion is based on the first subsystem composition of the FTC environment components. The \mathcal{R}_{CIN} relation is one level above, since it sums the subsystem requirements \mathcal{R}_{Sp} and \mathcal{R}_{CP} . To solve this problem equivalent classes that are at the same hierarchical level in one of the successive compositions of the system components are merged together. Hence, \mathcal{X} and \mathcal{U}_p , and $\tilde{\mathcal{X}}$ and $\tilde{\mathcal{U}}_p$ are combined to form the following common signature for \mathcal{R}_{Sp} and \mathcal{R}_{CP} :

$$\mathcal{R}_{Sign2} = ((\mathcal{X}, \mathcal{U}_p), (\tilde{\mathcal{X}}, \tilde{\mathcal{U}}_p)) \quad (4.17)$$

Table C.1 collects the original signature and the expanded signatures for each relational term used in the specification.

4.3 Formal specification of the FTC environment

Appendix B collects the elementary relational specifications of the FTC environment. More specifically, Section B.1 collects the elementary requirements of the AFCS performance specification, Section B.2 collects the elementary requirements of the DHC-2 detail-specification, Section B.3 collects the fault modes for each fallible component of the FTC environment, and Section B.4 collects the *restriction sets* used to delimit the scope of the AFCS specification when applied to the DHC-2 airplane.

Each relational requirement is denoted by the letter \mathcal{R} and a subscript that identifies the specific requirement (e.g. \mathcal{R}_{PAH} identifies the relational requirement for the PAH control function). Each requirement is formalized as illustrated in Section 4.1. The signature reported within the definition of the relation is the signature obtained after the first expansion discussed in the previous section. Table C.1 collects all elementary requirements along with a short explanation of the property captured by the relation, domain and image variables, and the signatures obtained after each expansion. The composition of the elementary specifications to form the performance and detail specification are discussed in the sections 4.3.1 and 4.3.2.

Airplane failure states are part of the DHC-2 detail-specification. Each airplane failure state consists of the airplane normal state modified by one or more malfunctions in airplane components or systems. Each mode of failure that is not extremely remote should be considered (see Section 3.1.6.2 of [4]). The specification of the detailed set of airplane failure states is out of the scope of this research. Hence, only one fault-mode for each fallible component is specified. Relations are used to

describe classes of fault-modes for a specific component. For example, the relation specifying the partial loss of rudder control surface models any partial loss up to total loss of the surface. The identifier of a fault-mode relation is the same as that used for the relation describing the required behaviour of the corresponding component, with a number added to the subscript to identify the fault mode (e.g. $\mathcal{R}_{p,1}$ is the relation describing fault mode number 1 for the roll rate gyro). The signature of a fault-mode relation is the same used in the relation specifying the requirements for the corresponding component, while the related predicate captures the specific faulty behaviour. The considered faults are very simple: these are partial loss of rudder control surface, engine loss, bad connection of sensor output, and stuck actuators. Sensor fault modes are an extrapolation of results from fault-mode-tests performed on rate gyros similar to the gyros adopted in this specification. For lack of aerodynamic data failure of a single aileron or a of single elevator are not modeled. Section B.3 collects all fault-mode relations, while Table C.3 lists them along with a brief description.

The *restriction-sets* are used to delimit the scope of the performance specification. There are three restriction-sets: \mathcal{S}_{ao} , \mathcal{S}_{at} , and \mathcal{S}_{Ctr} . \mathcal{S}_{ao} specifies the control-function selection logic; this information constitutes part of the airplane detail-specification as required in Section 3.1.2 of [2]. \mathcal{S}_{at} specifies the turbulence model to be adopted for verifying performance requirements with the airplane operating in turbulence. Only *random* turbulence is considered, while *discrete* turbulence (e.g. wind-gusts) and mean wind are not modeled. Effects of *wind-shear* are omitted since they become dominant in terminal flight-phases, while here the focus is on

autopilot functions that are engaged in non-terminal flight-phases only. The model adopted for random turbulence is the Dryden model with length and intensities scales as specified in Section 3.7 of [4]. $\mathcal{S}_{C_{tr}}$ represents the constraint over trim condition to guarantee a valid set of values. Each of these sets contains only a subset of the variables used within the performance specification. To make these sets homogeneous with the spaces of the relations they are going to be used with they are expanded as illustrated in Section 4.2. Table B.4 lists the restriction-sets along with a short description, the relevant variables, and the expanded spaces.

Appendix C collects all supporting tables for the FTC environment specification. Some of these tables have been already introduced, such as Table C.1 listing the elementary requirements of the FTC environment, Table C.3 listing the fault-mode relations, and Table C.4 listing the restriction sets used within the AFCS performance specification. Table C.2 lists the composed relations used within the performance and the detail specification along with domain and image spaces. Table C.5 lists all the classes of equivalent variables obtained with the first expansion discussed in Section 4.2 along with the set of variables belonging to each class. Table C.6 lists all domain and image variables used within the relational specification along with their data-type and a brief description of the represented quantities. Table C.7 defines the type and value of all the constants used within the specification. Table C.8 lists all quantified variables used within the specification along with their data-type. Table C.9 defines all the auxiliary terms, functions, and predicates. Finally, Table C.10 defines all the data-types. SI unit and range of allowed values is specified for

each data-type; this information can be used within the validation process to check requirement consistency. Value range is used to capture *indicative* constraint over requirement variables. For example, ranges for airspeed, altitude, and angle of attack are specified to capture the envelope of validity of the aerodynamic model of the airplane. Analogously, the rudderDeflection-T type specifies minimum and maximum deflection of the rudder; bankReference-T and pitchReference-T types specify the minimum and maximum values of bank and pitch reference angles as specified by the AFCS designer. Whenever strong typing of variables or constants would result in an excess of notation or in a cumbersome representation of data we omitted specification of type or the a looser form of typing was adopted where ranges and units are not specified. This is for example the case of the matrices of the actuator models in Table C.7 and of the stabilityDerivative-T and controlDerivative-T types in Table C.10.

4.3.1 Performance requirement composition

Section B.1 collects the elementary requirements for the AFCS performance specification. Of the performance specification for AFCS the author selected only those related to static accuracy for operation in smooth air and in turbulence that apply to the DHC-2 autopilot functions. These are the performance requirements for the PAH, ALH, RAH, HH, and HS autopilot functions, and the coordination requirements for lateral-directional control functions, both in Steady Banked Turn (SBT) and in Level Flight (LF). The original text of the requirements from [2] is reported along with the relational specification. An *intermediate* formulation of the requirement bridges the gap between the two specifications as illustrated in Section 4.1 for the HH function

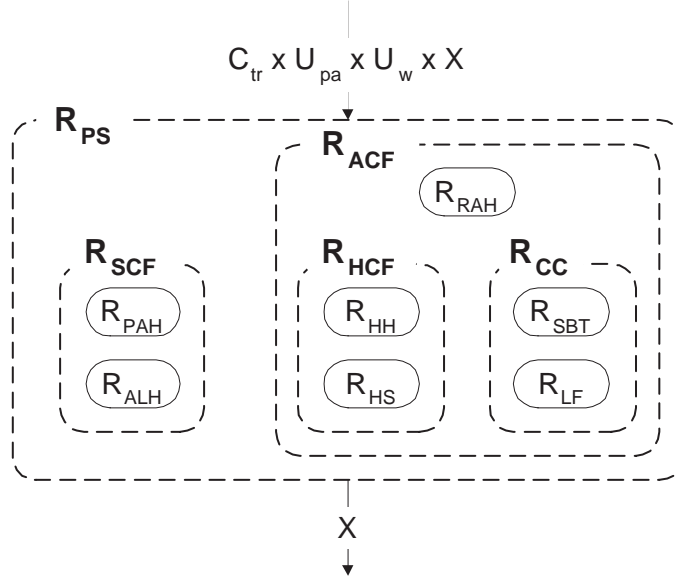


Figure 4.2: Structure of the AFCS performance requirements specification.

requirement. Requirements are numbered as in the original document; the ellipsis '...' is used to indicate the omission of a substantial part of the original text. The plain-English specification is divided into subsections marked with letters. These letters are used within the intermediate specification to point out corresponding subsections of the two formulations of the requirement; ambiguous subsections of the original text have been omitted from the formal specification.

Figure 4.2 represents the structure of the performance requirements. This structure is very simple; all requirements are combined by the join operator to form the Performance Specification \mathcal{R}_{PS} . The blocks with dash outline group the relations that specify the requirements for the Symmetric Control Functions \mathcal{R}_{SCF} , the relations that specify the Coordination Constraint \mathcal{R}_{CC} , the relations that specify the Heading Control Functions \mathcal{R}_{HCF} , and the relations that specify requirements for

the Asymmetric Control Functions \mathcal{R}_{ACF} . The performance specification \mathcal{R}_{PS} is a relation over the domain space $\mathcal{C}_{tr} + \mathcal{U}_{pa} + \mathcal{X} + \mathcal{U}_w$ and the image space \mathcal{X} . These are the expanded spaces that represent the class of trim-variables \mathcal{C}_{tr} , the class of AFCS input variables \mathcal{U}_{pa} , the class of air-turbulence variables \mathcal{U}_w , and the class of airplane state variables \mathcal{X} . Each requirement is expressed as a constraint over the evolution of the airplane state under specified turbulence conditions, for a specific AFCS input and trim condition.

The composition of the relations specifying the elementary performance requirements is straightforward; all requirements are combined through the join operator to form the performance specification. The resulting requirement is restricted to the admissible combination of AFCS inputs, to a valid set of trim-conditions, and to the required model of wind turbulence by means of the pre-restriction sets introduced in the previous section. The composition is expressed by the following equations:

$$\mathcal{S}_{PS} = \mathcal{S}_{C_{tr}} \cap \mathcal{S}_{ao} \cap \mathcal{S}_{at} \quad (4.18)$$

$$\mathcal{R}_{PS} = \mathcal{S}_{PS} \setminus \left(\mathcal{R}_{PAH} \sqcup \mathcal{R}_{ALH} \sqcup \mathcal{R}_{RAH} \sqcup \mathcal{R}_{HH} \sqcup \mathcal{R}_{HS} \sqcup \mathcal{R}_{SBT} \sqcup \mathcal{R}_{LF} \right) \quad (4.19)$$

4.3.2 DHC-2 detail-specification

Section B.2 collects the elementary requirements for the DHC-2 detail-specification. These relations specify the required behaviour of each component of the DHC-2 airplane and its AFCS. Since most of this information is in terms of analytical equations describing the operation of the corresponding component there is no plain-English

specification counterpart.

Elementary specifications are grouped into different subsections according to their role within the specification structure. Section B.2.1 collects relations that specify the airplane dynamics through force \mathcal{R}_{Feq} , moment \mathcal{R}_{Meq} , kinematic \mathcal{R}_{Keq} , and navigation \mathcal{R}_{Neq} equations. The forcing terms for these equations are aerodynamics forces \mathcal{R}_{aef} and moments \mathcal{R}_{aem} exerted by the airframe, forces \mathcal{R}_{csf} and moments \mathcal{R}_{csm} exerted by the control surfaces, forces \mathcal{R}_{pf} and moments \mathcal{R}_{pm} exerted by the engine, and gravity \mathcal{R}_{grf} and wind \mathcal{R}_{wf} forces. This section also includes the relations that specify air-data quantities \mathcal{R}_{ad} and kinematic acceleration at crew station \mathcal{R}_{ka} . Section B.2.2 collects relations that specify the requirements for the hardware components of the DHC-2 AFCS. These include actuator specifications \mathcal{R}_{rud} , \mathcal{R}_{ail} , and \mathcal{R}_{elv} , all sensor specifications \mathcal{R}_p , \mathcal{R}_q , \mathcal{R}_r , etc. control panel specification \mathcal{R}_{CP} , and FCC interface card specifications \mathcal{R}_{ADC} , and \mathcal{R}_{DAC} . Section B.2.3 collects relations that specify the requirements for the components of the FCSw. These are the interface modules \mathcal{R}_{in} and \mathcal{R}_{out} and all the modules related to the FCL: $\mathcal{R}_{\widehat{PAH}}$, $\mathcal{R}_{\widehat{ALH}}$, $\mathcal{R}_{\widehat{RAH}}$, $\mathcal{R}_{\widehat{HS}}$, and $\mathcal{R}_{\widehat{HH}}$.

Figure 4.3 illustrates the structure of the detail-specification. This structure is similar to the block diagram in figure 3.3. The main difference consists in the expansion of the airplane system into the relations that describe its dynamics. The display panel has been removed from the AFCS since it is not used within the specification. The diagram also displays the classes of variables over which these relations are defined. Some of these classes have been already introduced; some other classes

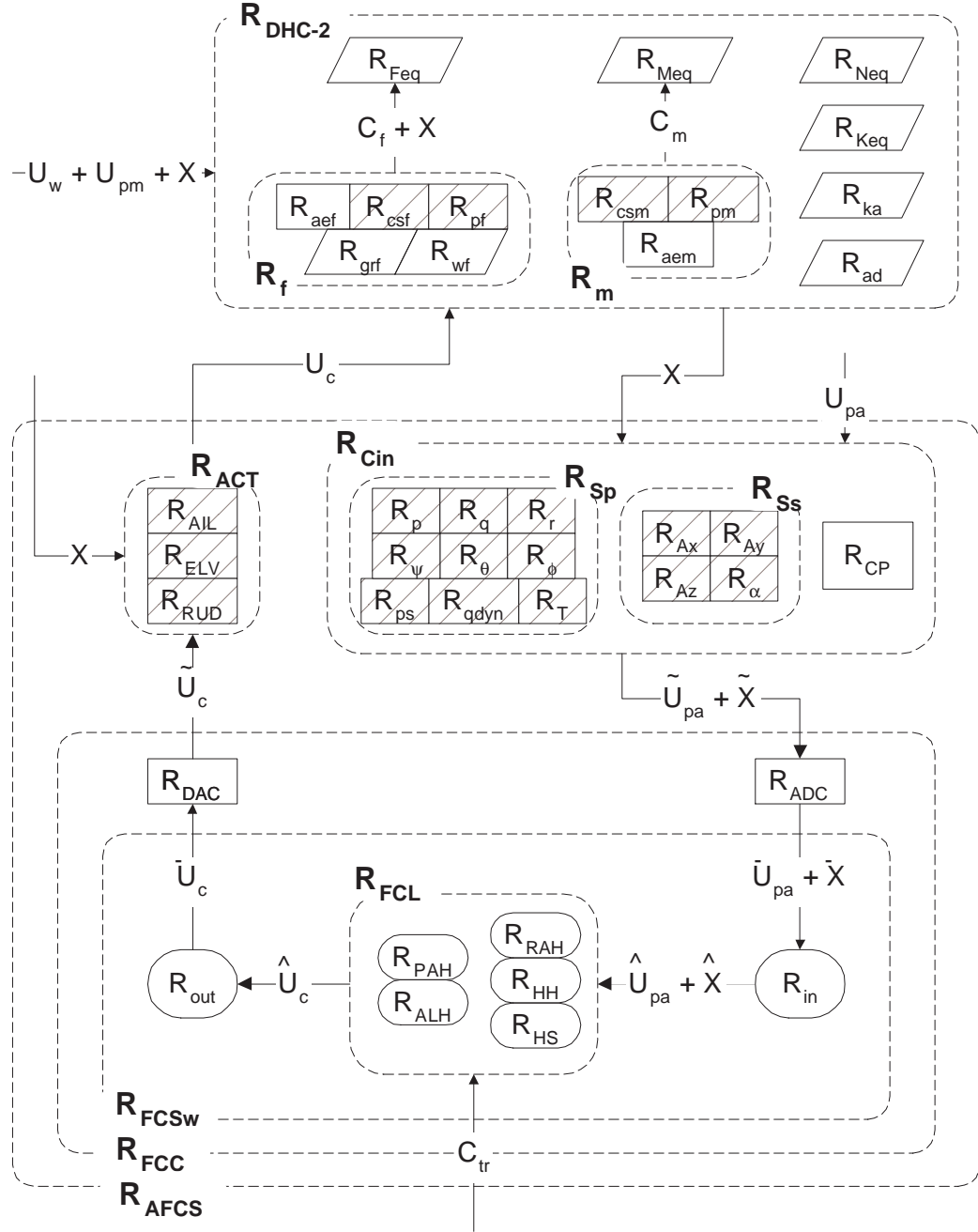


Figure 4.3: Structure of the DHC-2 detail-specification.

are introduced here for the first time. This is the case of the class of the MFCS control inputs \mathcal{U}_{pm} , the class of force and moment aerodynamic coefficients \mathcal{C}_f and \mathcal{C}_m , and the class of control surface deflection variables \mathcal{U}_c . Identifiers provided of a mathematical accent (*hat*, *tilde*, *bar*) represent quantities correlated to the quantity represented by the identifier without accent. The *tilde* accent is adopted for variables representing electrical signals, the *bar* accent is adopted for variables representing the ADC or DAC software representation of the quantity, and the *hat* accent is adopted for software variables. Hence, $\tilde{\mathcal{X}}$ is the class of variables representing electrical signals correlated to the airplane states, $\bar{\mathcal{X}}$ is the class of variables representing ADC software representation of the airplane states, and $\hat{\mathcal{X}}$ is the class of software variables representing the airplane states. Similar interpretations hold for the other accented classes and variables.

With the support of the specification structure in figure 4.3 the elementary requirements of Section B.2 can be composed to form the DHC-2 detail-specification. This specification is divided into the description of the DHC-2 airplane dynamics \mathcal{R}_{DHC2} and the DHC-2 AFCS \mathcal{R}_{AFCS} . Some relations are introduced to collect functionally related specifications. These are the relations \mathcal{R}_f and \mathcal{R}_m that collect respectively force and moment components involved in the flight dynamics equations, the relations \mathcal{R}_{S_p} and \mathcal{R}_{S_s} that represent the subsystem of primary and secondary sensors respectively, the relation \mathcal{R}_{FCL} that collects the flight control laws, the relation \mathcal{R}_{ACT} that collects the actuators, the relation \mathcal{R}_{FCSw} that represents the flight control software, and the \mathcal{R}_{FCC} that represents the flight control computer.

The formal composition of the elementary requirements is expressed by the following equations

$$\mathcal{R}_f = \mathcal{R}_{aef} \sqcup \mathcal{R}_{grf} \sqcup \mathcal{R}_{csf} \sqcup \mathcal{R}_{pf} \sqcup \mathcal{R}_{wf} \quad (4.20)$$

$$\mathcal{R}_m = \mathcal{R}_{aem} \sqcup \mathcal{R}_{csm} \sqcup \mathcal{R}_{pm} \quad (4.21)$$

$$\begin{aligned} \mathcal{R}_{DHC2} = & \mathcal{R}_f \circ \mathcal{R}_{Feq} \sqcup \mathcal{R}_m \circ \mathcal{R}_{Meq} \sqcup \mathcal{R}_{Keq} \\ & \sqcup \mathcal{R}_{Neq} \sqcup \mathcal{R}_{ad} \sqcup \mathcal{R}_{ka} \end{aligned} \quad (4.22)$$

$$\begin{aligned} \mathcal{R}_{S_p} = & \mathcal{R}_p \sqcup \mathcal{R}_q \sqcup \mathcal{R}_r \sqcup \mathcal{R}_\psi \sqcup \mathcal{R}_\theta \sqcup \mathcal{R}_\phi \\ & \sqcup \mathcal{R}_{p_s} \sqcup \mathcal{R}_{q_{dyn}} \sqcup \mathcal{R}_T \end{aligned} \quad (4.23)$$

$$\mathcal{R}_{S_s} = \mathcal{R}_{A_x} \sqcup \mathcal{R}_{A_y} \sqcup \mathcal{R}_{A_z} \sqcup \mathcal{R}_\alpha \quad (4.24)$$

$$\mathcal{R}_{FCL} = \mathcal{R}_{\widehat{PAH}} \sqcup \mathcal{R}_{\widehat{ALH}} \sqcup \mathcal{R}_{\widehat{RAH}} \sqcup \mathcal{R}_{\widehat{HH}} \sqcup \mathcal{R}_{\widehat{HS}} \quad (4.25)$$

$$\mathcal{R}_{ACT} = \mathcal{R}_{ail} \sqcup \mathcal{R}_{elv} \sqcup \mathcal{R}_{rud} \quad (4.26)$$

$$\mathcal{R}_{FCSw} = \mathcal{R}_{in} \circ \mathcal{R}_{FCL} \circ \mathcal{R}_{out} \quad (4.27)$$

$$\mathcal{R}_{FCC} = \mathcal{R}_{ADC} \circ \mathcal{R}_{FCSw} \circ \mathcal{R}_{DAC} \quad (4.28)$$

$$\mathcal{R}_{AFCS} = \left(\mathcal{R}_{S_p} \sqcup \mathcal{R}_{S_s} \sqcup \mathcal{R}_{CP} \right) \circ \mathcal{R}_{FCC} \circ \mathcal{R}_{ACT} \quad (4.29)$$

4.3.3 Correctness of AFCS design

The AFCS performance specification \mathcal{R}_{PS} represents the system requirements for an AFCS in terms of required behaviour of the airplane when equipped with the AFCS. The DHC-2 detail-specification and the AFCS performance specification can be composed as follows to form the specification of the airplane equipped with the AFCS:

$$\begin{aligned}
\mathcal{R}_{DHC2} \otimes \mathcal{R}_{AFCS} = & \tag{4.30} \\
& \left\{ \left((\mathcal{C}_{tr}, \mathcal{U}_w, \mathcal{U}_{pm}, \mathcal{U}_{pa}, \mathcal{X}), \mathcal{X}' \right) \middle| \exists \mathcal{U}_c \left(\right. \right. \\
& \quad \left((\mathcal{C}_{tr}, \mathcal{U}_{pa}, \mathcal{X}), \mathcal{U}_c \right) \in \mathcal{R}_{AFCS} \wedge \\
& \quad \left. \left. \left((\mathcal{U}_c, \mathcal{U}_w, \mathcal{U}_{pm}, \mathcal{X}), \mathcal{X}' \right) \in \mathcal{R}_{DHC2} \right) \right\}
\end{aligned}$$

The design specification of the AFCS is correct with respect to \mathcal{R}_{PS} if

$$\mathcal{R}_{DHC2} \otimes \mathcal{R}_{AFCS} \sqsubseteq \mathcal{R}_{PS} \tag{4.31}$$

Chapter 5

Formal requirements specification of the FTC

This chapter contains the formal specification of the FTC system. In the first section fault tolerance requirements are divided into functional and non-functional requirements and formulated in a form that facilitates their formalization. The relational specification is developed on top of the structured plain-English requirements. In the second section the focus is on the core of the FTC system, that is the FTC-AR module. The other FTC modules serve as interface and their formal specifications are collected in section B.5. The last section illustrates some concepts related to the feasibility analysis of the fault tolerance requirements.

5.1 FTC requirements

5.1.1 FTC functional requirements

The relation \mathcal{R}_{PS} captures the performance specification of the AFCS. If each component of the DHC-2 airplane equipped with AFCS satisfies the corresponding requirements then the correctness of the design, as formulated by equation 4.31, guarantees that the actual behaviour of the system satisfies the performance requirements. Fault

tolerance requirements aim to guarantee that performance specification are met also in case one or more of the system components fail to meet the corresponding specification. The main concepts related to fault tolerance for flight control systems were introduced in Section 3.1.2. The definition of fail operational and fail passive are reviewed here in support of the analysis that will lead to the formulation of the fault tolerance requirements for the AFCS of the DHC-2 airplane:

Fail operational The capability of the FCS for continued operation without degradation following a single failure, and to fail passive in the event of a related subsequent failure.

Fail passive The capability of the FCS to automatically disconnect and to revert to a passive state following a failure.

In the definition of fail operational capability the term *related subsequent failure* is ambiguous. In the first instance the requirement aims to capture the desired behaviour of the fault tolerant system under the condition of single and double failure. Hence, the temporal sequence in which the failures occur is not relevant. In the second instance, it is not clear under what conditions two failures are *related*. In the framework of physical redundancy the failure of two out of three redundant units might be considered as a case of related failures. However, this definition depends on the specific approach to achieving fault tolerance and loses its validity in the framework of analytical redundancy. The author adopts a definition that has a functional basis. Fallible components are partitioned into the following classes of equivalence: class of *measurement units* \mathcal{M} , class of *actuation units* \mathcal{A} , class of *processing units* \mathcal{P} , and class of *interface units* \mathcal{I} . Failures of components that belong to the same class are

considered related. This definition captures the author's best approach-independent interpretation of the term *related failures* in the original requirement. The rationale behind this interpretation is the implicit assumption in the definition of fail operational capability that the occurrence of two related failures is more critical than the occurrence of two unrelated failures. In fact, for related failures the fault tolerant system is required to be fail passive, while for unrelated failures it is required to continue operation. Failure of components serving the same function is potentially more critical than failure of components serving different functions. Hence, the component function is a suitable basis for defining *related* failures.

Proper formulation of the automatic disengagement requirement as it appears in the fail passive definition would require the formal description of the MFCS. The details of the transition from AFCS to MFCS following automatic disengagement are not considered; the FTC is simply required to signal this transition.

Another requirement for fault tolerant flight control systems is the "*FCS warning and status annunciation*" requirement (section 3.2.1.4.2 of [2]). The FTC is required to signal a warning in correspondence of a fault and of automatic disengagement. Hence, the FTC shall have a display panel with a warning light for each monitored component and an additional light for automatic disengagement. Furthermore, the FTC control panel shall have a switch to control the ON/OFF status of the FTC system.

The fail operational capability definition and the warning and status annunciation requirement represent the core of the fault tolerance requirements specification. To

formulate the fault tolerance requirements in a format more suitable to formalization the following fault hypotheses are introduced:

\mathcal{H}_0 Fault free hypothesis; all of the components used by the FTFCS are working according to the corresponding specifications.

\mathcal{H}_1 Single failure hypothesis; at least one component among those used by the FTFCS is working according to one of its fault modes, and there are no related faults;

\mathcal{H}_2 Multiple failure hypothesis; at least two related components used by the FTFCS are working according to one of the corresponding fault modes.

In the definition of fault hypotheses \mathcal{H}_1 and \mathcal{H}_2 the fault modes have been adopted to specify the behaviour of a component that is not working according to its specification. The rationale behind this choice is that required behaviour and faulty behaviours do not necessarily cover all the possible modes of operation of a component. Hence, both the required behaviour and the set of fault modes are adopted as basis for the certification, as opposed to adopting the required behaviour and any possible behaviour that does not refine the component specification.

The fault tolerance requirements are formulated as follows:

- if the FTC is not engaged then the FTFCS shall operate like the original AFCS and all FTC warning lights shall be off;
- if the FTC is engaged then
 - under conditions captured by the fault hypothesis \mathcal{H}_0 the DHC-2 airplane equipped with the FTFCS shall meet the performance specification \mathcal{R}_{PS} and all FTC warning lights shall be off;

- under conditions captured by the fault hypothesis \mathcal{H}_1 the DHC-2 airplane equipped with the FTFCS shall meet the performance specification \mathcal{R}_{PS} and only the warning lights corresponding to the faulty components shall be on;
- under conditions captured by the fault hypothesis \mathcal{H}_2 the FTC automatic-disengagement warning light shall be on.

5.1.2 FTC non-functional requirements

In Section 3.1.2 two non-functional requirements for the FTC system were introduced. These requirements were in the form of constraint over the modular architecture of the solution and over the redundancy approach to be adopted for each component. According to this last requirement fault tolerance must be achieved by means of physical or analytical redundancy depending on the nature of the component. Fault tolerance with respect to failure of components that belongs either to class \mathcal{P} or to class \mathcal{I} must be achieved adopting the physical redundancy approach, while fault tolerance with respect to failure of components of classes \mathcal{M} and \mathcal{A} must be achieved adopting the analytical redundancy approach. Furthermore, the solution must be modular, with the following assigned priority among modules: fault tolerance with respect to failure of components of class \mathcal{P} and \mathcal{I} is achieved first, then fault tolerance with respect to failure of components of class \mathcal{M} , and finally fault tolerance with respect to failure of components of class \mathcal{A} . This priority ordering implies that the failure of components of class \mathcal{P} and \mathcal{I} is transparent to the module that is required to provide fault tolerance with respect to failure of components of class \mathcal{M} , while the failure of components of class \mathcal{A} is not. For this research the focus is on the FTC

module that provides fault tolerance with respect to sensor failures.

The above non-functional constraints have an impact on the specification of the FTC requirements. The analytical redundancy approach constraint affects the fault hypothesis definitions. Since analytical redundancy maps into the correlation of sensor outputs and actuator inputs at software level, each failure that might affect such correlation must be considered; that is, fault hypotheses must be total with respect to operation of fallible components. The module priority constraint also has an impact on the definition of the fault hypotheses, since it determines whether a failure should be considered transparent to the FTC module or not. Furthermore, the modular approach translates into a shift in the target of the FTC requirements from preserving AFCS performance specification in case of single failure to preserving functionality of the set of sensors. The specification of the set of sensors is:

$$\mathcal{R}_{X\hat{X}} = (\mathcal{R}_{Sp} \sqcup \mathcal{R}_{Ss}) \circ \mathcal{R}_{ADC} \circ \mathcal{R}_{in} \quad (5.1)$$

5.1.3 FTC-AR requirements

Figure 5.1 represents the specification structure of the DHC-2 airplane equipped with the FTFCS. Blocks with a thicker outline represent the components of the FTC system. The core of the FTC is the FTC-AR module. This is the component required to exploit analytical redundancy of the DHC-2 system to provide fault tolerance with respect to sensor failures. The other components of the FTC system are the control (CP) and display (DP) panels, that serve as interface between the FTC and the pilot, the hardware (ADC, DAC) and software blocks (IN, OUT) that link the FTC external interface to the core module FTC-AR, and the FTC-SW block that serves as a switch

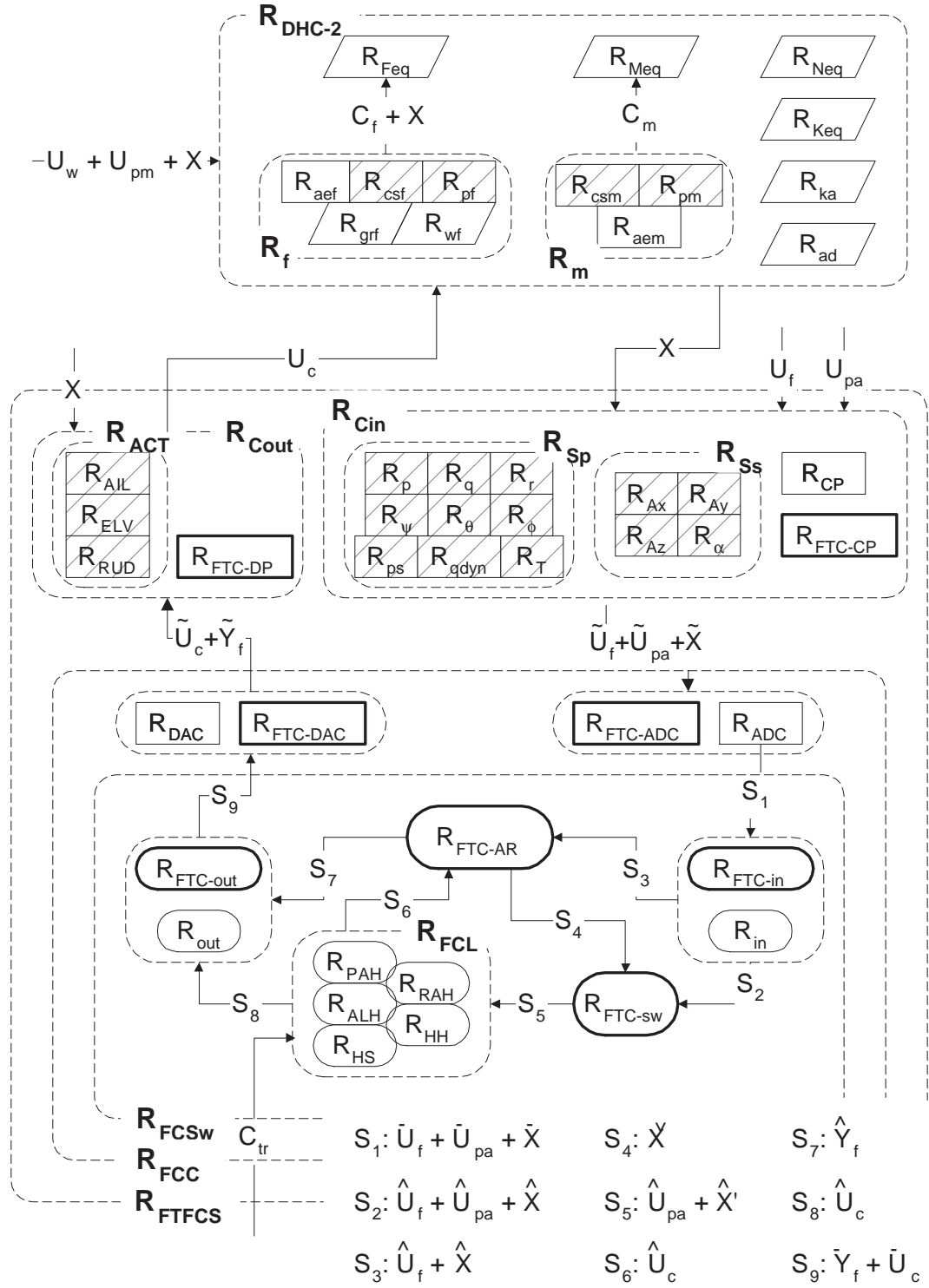


Figure 5.1: DHC-2 and FTFCS requirements specification structure.

between the original AFCS and the FTFCS. The FTC-SW fulfills a safety function; if the FTC is disengaged the FTC-SW block isolates the FTC-AR module so that its output does not affect the input to the FCL. Introduction of the FTC system into the AFCS leads to the introduction of new quantities used in the formulation of the requirements. These quantities are represented by the class of FTC input variables \mathcal{U}_f , the class of FTC warning variables \mathcal{Y}_f , and the class of validated airplane states $\tilde{\mathcal{X}}$.

With the decomposition of the FTC system into its elementary components the FTC-AR module has been isolated. This is the FTC module that inherits the fault tolerance requirements. The FTC-AR requirements are derived from the fault tolerance requirements after integrating them with the non-functional constraints and by projecting them onto its interface. This process leads to the following fault hypotheses:

\mathcal{H}_{M0} Fault free hypothesis;

- all of the sensors used by the FTFCS are working according to the corresponding specifications and
- all of the processing and interface units used by the FTFCS are working according to the corresponding specifications and
- no more than two actuation units used by the FTFCS are working according to one of the corresponding fault modes, while all the others are working according to the corresponding specification, and
- all fallible components not used by the FTFCS are either working according to the corresponding specification or according to one of the corresponding fault modes.

$\mathcal{H}_{M1,m}$ Single failure hypothesis;

- the sensor used by the FTFCS and identified by subscript m is working according to one of its fault modes while all the others are working according to the corresponding specifications and
- hypothesis among remaining fallible components: same as for \mathcal{H}_{M0}

\mathcal{H}_{M2} Multiple failure hypothesis;

- at least two sensors used by the FTFCS are working according to one of the corresponding fault modes while all the others are working according to the corresponding specifications and
- hypothesis among remaining fallible components: same as for \mathcal{H}_{M0}

The fault tolerance requirements translate into:

- a) if \widehat{SW}_{FTC} is OFF then all warning variables shall be OFF;
- b) if \widehat{SW}_{FTC} is ON then
 - b1) if the correlation between $\hat{\mathcal{U}}_c$ and $\hat{\mathcal{X}}$ corresponds to the status described by fault hypothesis \mathcal{H}_{M0} then all FTC warning variables shall be OFF;
 - b2) if the correlation between $\hat{\mathcal{U}}_c$ and $\hat{\mathcal{X}}$ corresponds to the status described by fault hypothesis $\mathcal{H}_{M1,m}$ then \hat{W}_m shall be ON, and all other FTC warning variables shall be OFF;
 - b3) if the correlation between $\hat{\mathcal{U}}_c$ and $\hat{\mathcal{X}}$ corresponds to the status described by fault hypothesis \mathcal{H}_{M2} then \hat{W}_{dis} shall be ON.
 - b4) if the correlation between $\hat{\mathcal{U}}_c$ and $\hat{\mathcal{X}}$ corresponds to the status described by fault hypothesis \mathcal{H}_{M0} or by any of the fault hypotheses $\mathcal{H}_{M1,m}$ then the set of sensors equipped with the FTC shall meet the specification $\mathcal{R}_{\mathbf{X}\hat{\mathbf{X}}}$

Fault tolerance requirements are decomposed into fault-free requirement (item a), detection and identification requirements (items b1, b2, and b3), and recovery requirement (item b4). The rationale behind this choice is the difference in the requirements signatures. Detection and identification requirements are expressed in terms of FTC-AR inputs and outputs, while the recovery requirement is expressed in terms of the actual airplane states and their corresponding software variables.

5.2 Formal specification of FTC-AR

Section B.5 collects the relations that specify the requirements of the FTC-AR interface blocks. Interpretation of these relations is straightforward; these blocks are similar to the corresponding interface blocks of the AFCS relational specification.

The relational specification for the FTC-AR represents the formalization of the fault-tolerance requirements developed in the previous section. The relational specification is structured like the plain-English specification, with fault hypotheses serving as support to the actual requirements specification.

Before developing the relational specification of the FTC-AR requirements the partitioning among the set of components must be made explicit and the fault hypotheses must be formalized.

5.2.1 Components partitioning

The DHC-2 airplane equipped with the FTFCS has been decomposed in components, and each component has been assigned a relational specification as shown in figure 5.1. The components themselves are identified by the letter C with a subscript identical

to that of the corresponding relation. The set of all components is \mathcal{C} , the set of components used by the FTFCS is \mathcal{U} , the set of components that can fail is \mathcal{F} . The latter set is partitioned into the subset of measurement components \mathcal{M} , the subset of actuation components \mathcal{A} , the subset of processing components \mathcal{P} , and the subset of interface components \mathcal{I} .

Set of components

$$\begin{aligned} \mathcal{C} = \{ & C_{Feq}, C_{Meq}, C_{Keq}, C_{Neq}, \\ & C_{aef}, C_{aem}, C_{grf}, C_{wff}, C_{ka}, C_{ad}, C_{csf}, C_{csm}, C_{pf}, C_{pm}, \\ & C_{rud}, C_{ail}, C_{elv}, \\ & C_p, C_q, C_r, C_\theta, C_\phi, C_\psi, C_{qdyn}, C_{ps}, C_T, \\ & C_{Ax}, C_{Ay}, C_{Az}, C_\alpha, \\ & C_{CP}, C_{DAC}, C_{ADC}, \\ & C_{in}, C_{out}, C_{\widehat{PAH}}, C_{\widehat{ALH}}, C_{\widehat{RAH}}, C_{\widehat{HH}}, C_{\widehat{HS}}, \\ & C_{FTC-CP}, C_{FTC-DP}, C_{FTC-ADC}, C_{FTC-DAC}, \\ & C_{FTC-IN}, C_{FTC-OUT}, C_{FTC-SW}, C_{FTC-AR} \} \end{aligned} \quad (5.2)$$

Set of components used within the FTFCS

$$\begin{aligned} \mathcal{U} = \{ & C_{csf}, C_{csm}, C_{pf}, C_{pm}, \\ & C_{rud}, C_{ail}, C_{elv}, \\ & C_p, C_q, C_r, C_\theta, C_\phi, C_\psi, C_{qdyn}, C_{ps}, C_T, \\ & C_{CP}, C_{DAC}, C_{ADC}, \\ & C_{in}, C_{out}, C_{\widehat{PAH}}, C_{\widehat{ALH}}, C_{\widehat{RAH}}, C_{\widehat{HH}}, C_{\widehat{HS}}, \\ & C_{FTC-CP}, C_{FTC-DP}, C_{FTC-ADC}, C_{FTC-DAC}, \\ & C_{FTC-IN}, C_{FTC-OUT}, C_{FTC-SW}, C_{FTC-AR} \} \end{aligned}$$

Set of fallible components

$$\begin{aligned}
\mathcal{F} = \{ & C_{csf}, C_{csm}, C_{pf}, C_{pm}, \\
& C_{rud}, C_{ail}, C_{elv}, \\
& C_p, C_q, C_r, C_\theta, C_\phi, C_\psi, C_{qdyn}, C_{ps}, C_T, \\
& C_{Ax}, C_{Ay}, C_{Az}, C_\alpha, \\
& C_{CP}, C_{DAC}, C_{ADC}, \\
& C_{in}, C_{out}, C_{\widehat{PAH}}, C_{\widehat{ALH}}, C_{\widehat{RAH}}, C_{\widehat{HH}}, C_{\widehat{HS}}, \\
& C_{FTC-CP}, C_{FTC-DP}, C_{FTC-ADC}, C_{FTC-DAC}, \\
& C_{FTC-IN}, C_{FTC-OUT}, C_{FTC-SW}, C_{FTC-AR} \}
\end{aligned} \tag{5.3}$$

Set of Measurement components

$$\begin{aligned}
\mathcal{M} = \{ & C_p, C_q, C_r, C_\theta, C_\phi, C_\psi, C_{qdyn}, C_{ps}, C_T, \\
& C_{Ax}, C_{Ay}, C_{Az}, C_\alpha \}
\end{aligned} \tag{5.4}$$

Set of Actuation components

$$\mathcal{A} = \{ C_{csf}, C_{csm}, C_{pf}, C_{pm}, C_{rud}, C_{ail}, C_{elv} \} \tag{5.5}$$

Set of Processing components

$$\begin{aligned}
\mathcal{P} = \{ & C_{\widehat{ALH}}, C_{\widehat{RAH}}, C_{\widehat{HH}}, C_{\widehat{HS}}, \\
& C_{FTC-IN}, C_{FTC-OUT}, C_{FTC-SW}, C_{FTC-AR} \}
\end{aligned} \tag{5.6}$$

Set of Interface components

$$\begin{aligned}
\mathcal{I} = \{ & C_{CP}, C_{DAC}, C_{ADC}, \\
& C_{FTC-CP}, C_{FTC-DP}, C_{FTC-ADC}, C_{FTC-DAC} \}
\end{aligned} \tag{5.7}$$

5.2.2 Formal specification of fault hypotheses

A fault hypothesis consists in the assignment to each fallible component of a mode of operation that is either described by the component specification or by one of the component fault-modes. For each component \mathcal{C}_c , \mathcal{R}_c denotes the relation that specifies its requirement, \mathcal{F}_c denotes the set of its fault-mode relations, and $\mathcal{R}_{c,n}$ denotes the relation that specifies its fault-mode number n . $\mathcal{R}_c(\mathcal{H})$ denotes the behaviour of component \mathcal{C}_c under fault hypothesis \mathcal{H} and the term \mathcal{H}_* denotes any of the fault hypotheses. For the sake of space the definition of fault hypothesis \mathcal{H}_{M0} is explicit, while the other fault hypothesis definitions refer to this one.

Fault hypothesis \mathcal{H}_{M0}

Measurement components used by the FTFCS

$$\forall m \left(\mathcal{C}_m \in \mathcal{M} \cap \mathcal{U} \Rightarrow \mathcal{R}_m(\mathcal{H}_{M0}) = \mathcal{R}_m \right) \quad (5.8)$$

Processing components used by the FTFCS

$$\forall c \left(\mathcal{C}_c \in \mathcal{P} \cap \mathcal{U} \Rightarrow \mathcal{R}_c(\mathcal{H}_*) = \mathcal{R}_c \right) \quad (5.9)$$

Interface components used by the FTFCS

$$\forall c \left(\mathcal{C}_c \in \mathcal{I} \cap \mathcal{U} \Rightarrow \mathcal{R}_c(\mathcal{H}_*) = \mathcal{R}_c \right) \quad (5.10)$$

Actuation components used by the FTFCS

$$\begin{aligned} & \exists a \exists f \exists b \exists g \left(\mathcal{C}_a \in \mathcal{A} \cap \mathcal{U} \wedge \mathcal{R}_{a,f} \in \mathcal{F}_a \wedge \right. \\ & \quad \mathcal{C}_b \in \mathcal{A} \cap \mathcal{U} \wedge \mathcal{R}_{b,g} \in \mathcal{F}_b \Rightarrow \\ & \quad \left(\mathcal{R}_a(\mathcal{H}_*) = \mathcal{R}_{a,f} \vee \mathcal{R}_a(\mathcal{H}_*) = \mathcal{R}_a \right) \wedge \\ & \quad \left(\mathcal{R}_b(\mathcal{H}_*) = \mathcal{R}_{b,g} \vee \mathcal{R}_b(\mathcal{H}_*) = \mathcal{R}_b \right) \wedge \\ & \quad \left. \forall c \left(c \neq a \wedge c \neq b \wedge \mathcal{C}_c \in \mathcal{A} \cap \mathcal{U} \Rightarrow \mathcal{R}_c(\mathcal{H}_*) = \mathcal{R}_c \right) \right) \end{aligned} \quad (5.11)$$

Components not used by the FTFCS

$$\begin{aligned} \forall c \left(\mathcal{C}_c \in \mathcal{F} \setminus \mathcal{U} \Rightarrow \mathcal{R}_c(\mathcal{H}_*) = \mathcal{R}_c \vee \right. \\ \left. \exists f \left(\mathcal{R}_{c,f} \in \mathcal{F}_c \Rightarrow \mathcal{R}_c(\mathcal{H}_*) = \mathcal{R}_{c,f} \right) \right) \end{aligned} \quad (5.12)$$

Fault hypothesis $\mathcal{H}_{M1,m}$

Measurement components used by the FTFCS

$$\begin{aligned} \exists f \left(\mathcal{C}_m \in \mathcal{M} \cap \mathcal{U} \wedge \mathcal{R}_{m,f} \in \mathcal{F}_m \Rightarrow \right. \\ \left. \mathcal{R}_m(\mathcal{H}_{M1}) = \mathcal{R}_{m,f} \right) \wedge \\ \forall n \left(n \neq m \wedge \mathcal{C}_n \in \mathcal{M} \cap \mathcal{U} \Rightarrow \mathcal{R}_n(\mathcal{H}_{M1}) = \mathcal{R}_n \right) \end{aligned} \quad (5.13)$$

All other fallible components: same as for fault hypothesis \mathcal{H}_{M0}

Fault hypothesis \mathcal{H}_{M2}

Measurement components used by the FTFCS

$$\begin{aligned} \exists m \exists f \exists n \exists g \left(\mathcal{C}_m \in \mathcal{M} \cap \mathcal{U} \wedge \mathcal{R}_{m,f} \in \mathcal{F}_m \wedge \right. \\ \left. \mathcal{C}_n \in \mathcal{M} \cap \mathcal{U} \wedge \mathcal{R}_{n,g} \in \mathcal{F}_n \Rightarrow \right. \\ \left. \mathcal{R}_m(\mathcal{H}_{M2}) = \mathcal{R}_{m,f} \wedge \mathcal{R}_n(\mathcal{H}_{M2}) = \mathcal{R}_{n,g} \wedge \right. \\ \left. \forall c \left(c \neq m \wedge c \neq n \wedge \mathcal{C}_c \in \mathcal{M} \cap \mathcal{U} \Rightarrow \right. \right. \\ \left. \left. \mathcal{R}_c(\mathcal{H}_{M2}) = \mathcal{R}_c \vee \right. \right. \\ \left. \left. \exists h \left(\mathcal{R}_{c,h} \in \mathcal{F}_c \Rightarrow \mathcal{R}_c(\mathcal{H}_{M2}) = \mathcal{R}_{c,h} \right) \right) \right) \end{aligned} \quad (5.14)$$

All other fallible components: same as for fault hypothesis \mathcal{H}_{M0}

5.2.3 Relational specification of the FTC-AR requirements

The relations of this section specify the fault tolerance requirements of section 5.1.3.

$\mathcal{R}_{ENV}(\mathcal{H})$ describes the environment of the FTC-AR block under fault hypothesis

\mathcal{H} ; this relation capture the actual correlation between software variables representing airplane state and actuator inputs. $\mathcal{R}_{FTC-OFF}$ specifies the requirements for FTC disengaged, \mathcal{R}_{FTC-DI} specifies the detection and identification requirements under all fault hypotheses, and \mathcal{R}_{FTC-R} specifies the recovery requirements.

$$\begin{aligned}
\mathcal{R}_{ENV}(\mathcal{H}) = & \quad (5.15) \\
& \left\{ \left(\hat{\mathcal{U}}_c, \hat{\mathcal{X}} \right) \middle| \exists \mathcal{U}'_w \exists \mathcal{U}'_{pm} \exists \mathcal{U}'_c \exists \tilde{\mathcal{U}}'_c \exists \mathcal{U}'_{pa} \exists \mathcal{X}' \left(\right. \\
& \quad \left(\hat{\mathcal{U}}_c, \tilde{\mathcal{U}}'_c \right) \in \mathcal{R}_{out}(\mathcal{H}) \circ \mathcal{R}_{DAC}(\mathcal{H}) \wedge \\
& \quad \left((\tilde{\mathcal{U}}'_c, \mathcal{X}'), \mathcal{U}'_c \right) \in \mathcal{R}_{ail}(\mathcal{H}) \sqcup \mathcal{R}_{elv}(\mathcal{H}) \sqcup \mathcal{R}_{rud}(\mathcal{H}) \wedge \\
& \quad \left((\hat{\mathcal{U}}'_c, \mathcal{U}'_{pm}, \mathcal{U}'_w, \mathcal{X}'), \mathcal{X}' \right) \in \\
& \quad \mathcal{R}_{Keq} \sqcup \mathcal{R}_{Neq} \sqcup \mathcal{R}_{ad} \sqcup \mathcal{R}_{ka} \sqcup \\
& \quad \left(\mathcal{R}_{aef} \sqcup \mathcal{R}_{grf} \sqcup \mathcal{R}_{csf}(\mathcal{H}) \sqcup \mathcal{R}_{pf}(\mathcal{H}) \sqcup \mathcal{R}_{wf} \right) \circ \mathcal{R}_{Feq} \sqcup \\
& \quad \left(\mathcal{R}_{aem} \sqcup \mathcal{R}_{csm}(\mathcal{H}) \sqcup \mathcal{R}_{pm}(\mathcal{H}) \right) \circ \mathcal{R}_{Meq} \wedge \\
& \quad \left((\mathcal{U}'_{pa}, \mathcal{X}'), \hat{\mathcal{X}} \right) \in \left(\mathcal{R}_p(\mathcal{H}) \sqcup \mathcal{R}_q(\mathcal{H}) \sqcup \mathcal{R}_r(\mathcal{H}) \sqcup \right. \\
& \quad \mathcal{R}_\psi(\mathcal{H}) \sqcup \mathcal{R}_\theta(\mathcal{H}) \sqcup \mathcal{R}_\phi(\mathcal{H}) \sqcup \mathcal{R}_{ps}(\mathcal{H}) \sqcup \mathcal{R}_{qdyn}(\mathcal{H}) \sqcup \mathcal{R}_T(\mathcal{H}) \sqcup \\
& \quad \left. \mathcal{R}_{Ax}(\mathcal{H}) \sqcup \mathcal{R}_{Ay}(\mathcal{H}) \sqcup \mathcal{R}_{Az}(\mathcal{H}) \sqcup \mathcal{R}_\alpha(\mathcal{H}) \sqcup \mathcal{R}_{CP}(\mathcal{H}) \right) \circ \\
& \quad \left. \mathcal{R}_{ADC}(\mathcal{H}) \circ \mathcal{R}_{in}(\mathcal{H}) \right\}
\end{aligned}$$

$$\begin{aligned}
\mathcal{R}_{FTC-OFF} = & \quad (5.16) \\
& \left\{ \left(\hat{\mathcal{U}}_f, \hat{\mathcal{Y}}_f \right) \middle| \right. \\
& \quad \forall k_1 \forall k_2 \left(0 \leq k_1 < k_2 < \infty \Rightarrow \right. \\
& \quad \forall k \left(k_1 < k < k_2 \wedge \widehat{SW}_{FTC}(k) = OFF \Rightarrow \right. \\
& \quad \left. \left. \forall m \left(\mathcal{C}_m \in \mathcal{M} \cap \mathcal{U} \Rightarrow \hat{W}_m(k) = OFF \right) \wedge \hat{W}_{dis}(k) = OFF \right) \right\}
\end{aligned}$$

$$\begin{aligned}
\mathcal{R}_{FTC-DI} = & \quad (5.17) \\
& \left\{ \left((\hat{\mathcal{U}}_f, \hat{\mathcal{U}}_c, \hat{\mathcal{X}})(\hat{\mathcal{Y}}_f) \right) \middle| \right. \\
& \quad \forall k_1 \forall k_2 \left(0 \leq k_1 < k_2 < \infty \Rightarrow \right. \\
& \quad \forall k \left(k_1 < k < k_2 \wedge \widehat{SW}_{FTC}(k) = ON \Rightarrow \right. \\
& \quad \left((\hat{\mathcal{U}}_c, \hat{\mathcal{X}}) \in \mathcal{R}_{ENV}(\mathcal{H}_{M0}) \wedge \right. \\
& \quad \left. \forall m \left(\mathcal{C}_m \in \mathcal{M} \cap \mathcal{U} \Rightarrow \hat{W}_m(k) = OFF \right) \wedge \right. \\
& \quad \left. \hat{W}_{dis}(k) = OFF \right) \vee \\
& \quad \exists c \left((\hat{\mathcal{U}}_c, \hat{\mathcal{X}}) \in \mathcal{R}_{ENV}(\mathcal{H}_{M1,c}) \wedge \right. \\
& \quad \forall m \left(\mathcal{C}_m \in \mathcal{M} \cap \mathcal{U} \wedge m \neq c \Rightarrow \hat{W}_m(k) = OFF \right) \wedge \\
& \quad \left. \hat{W}_{dis}(k) = OFF \wedge \hat{W}_c(k) = ON \right) \vee \\
& \quad \left. \left. \left((\hat{\mathcal{U}}_c, \hat{\mathcal{X}}) \in \mathcal{R}_{ENV}(\mathcal{H}_{M2}) \wedge \hat{W}_{dis}(k) = ON \right) \right) \right\}
\end{aligned}$$

$$\begin{aligned}
\mathcal{R}_{FTC-R} = & \quad (5.18) \\
& \left\{ \left((\mathcal{X}, \hat{\mathcal{X}}, \hat{\mathcal{U}}_c), \check{\mathcal{X}} \right) \middle| \forall m \left(\mathcal{C}_m \in \mathcal{M} \cap \mathcal{U} \wedge \right. \right. \\
& \quad \left. \left. (\hat{\mathcal{U}}_c, \hat{\mathcal{X}}) \in \mathcal{R}_{ENV}(\mathcal{H}_{M0}) \cup \mathcal{R}_{ENV}(\mathcal{H}_{M1,m}) \Rightarrow (\mathcal{X}, \check{\mathcal{X}}) \in \mathcal{R}_{X\hat{X}} \right) \right\}
\end{aligned}$$

5.3 Feasibility analysis

The relations of the previous section along with the elementary specification and tables collected in the appendices B and C represent the baseline of the requirements specification for the FTC system. This marks the end of the specification phase of the requirements engineering process; the next phase is related to the validation of the requirements. This phase requires the support of automatic tools to check for consistency of the specification, and a team of domain experts to evaluate specification

completeness and correctness. This is beyond the scope of this research; however, the feasibility aspect of the validation phase is discussed in some detail. Feasibility of the fault tolerance requirements is related to detectability of a fault condition, identifiability of the faulty component, and recoverability of the function provided by the faulty component. The focus is on detectability and identifiability. Before analyzing these two items on the basis of the formal specification their interpretation in the technical literature is illustrated.

5.3.1 Traditional interpretation of detectability and identifiability

Fault detectability and identifiability (or isolability) are relatively new concepts in the field of fault diagnosis. Only a few papers in the technical literature focus on these concepts, and present slightly different interpretations. Frank et al. [24] define a system to be *unknown-input fault detectable* if for "*almost all faults, an arbitrary small time interval allows a unique decision for the fault only on considering the known input and the available output data in this time interval*". According to this definition fault detectability is a system property that holds for *almost all faults*. However, in a system there can be both detectable and undetectable faults at the same time; an example is provided in [29]. Hence, the above definition leads to too restrictive detectability conditions. In [32] Horak states that a fault is detectable if its effects on system outputs overtake the effects of model uncertainties. Using an optimization procedure based on the Maximum Principle, the maximum possible deviation between nominal output values and actual output values (*reachable mea-*

measurements interval) is derived at each time step. A fault is declared detectable if it causes the system output to assume values outside the reachable measurement interval. This definition of fault detectability highlights the concept of detection feasibility in spite of disturbances; however, it differentiates detectable from undetectable faults in terms of fault effects in time domain only. Faults that add to system output a low-amplitude, high-frequency component could be detected by analyzing the frequency content of the measurements. Nevertheless, they are undetectable, according to the definition in [32], if fault effects fall within the reachable measurement interval. In other definitions fault detectability is either intended as detection capability of a specific fault detection system [13], or as the best detection capability achievable by adopting a particular residual generation approach ([23] and [43]). In [11] (sections 2.6 and 2.7) fault detectability and isolability are defined in terms of the fault transfer matrix obtained from the adopted residual generator: a fault is detectable if the transfer function between the fault input and the residual vector is non-zero; a fault is identifiable if the residual vector allows distinguishing it from the other faults.

None of the above interpretations provide a definition that is independent of the specific residual generator adopted and that captures all relevant elements of the detectability and isolability problems. An attempt to provide a definition of detectability as a system property (i.e. independent of the adopted residual generator) and to point out the elements that play a role in the detectability problem can be found in [29]. In this article fault detectability is defined as follows: *"A fault on a component of the system is said to be detectable if knowledge of system inputs and outputs*

over a finite time interval following the occurrence of the fault allows the detection in spite of disturbances”. Fault dynamics, disturbance action, and system dynamics are pointed out as the only elements that play a role in assessing fault detectability. A detectability analytical criterion is derived from the above definition and formulated in the frequency domain.

5.3.2 Formal definition of detectability and identifiability

The definition of detectability and identifiability can be formulated on the basis of the fault detection and identification requirements specified in section 5.1.3:

Detectability condition

- the correlation between $\hat{\mathcal{U}}_c$ and $\hat{\mathcal{X}}$ allows distinguishing between conditions captured by fault hypothesis \mathcal{H}_{M0} and conditions captured by any of the fault hypotheses $\mathcal{H}_{M1,m}$ and
- the correlation between $\hat{\mathcal{U}}_c$ and $\hat{\mathcal{X}}$ allows distinguishing between conditions captured by fault hypothesis \mathcal{H}_{M0} and conditions captured by fault hypothesis \mathcal{H}_{M2} and
- the correlation between $\hat{\mathcal{U}}_c$ and $\hat{\mathcal{X}}$ allows distinguishing between conditions captured by any of the fault hypotheses $\mathcal{H}_{M1,m}$ and conditions captured by fault hypothesis \mathcal{H}_{M2} ;

Identifiability

- The correlation between $\hat{\mathcal{U}}_c$ and $\hat{\mathcal{X}}$ allows distinguishing between conditions captured by any couple of fault hypotheses $\mathcal{H}_{M1,m}$ and $\mathcal{H}_{M1,n}$

The above definition can be formalized as follows:

$$\forall m \left(\mathcal{C}_m \in \mathcal{M} \cap \mathcal{U} \Rightarrow \right. \quad (5.19)$$

$$\begin{aligned} & \mathcal{R}_{ENV}(\mathcal{H}_{M0}) \cap \mathcal{R}_{ENV}(\mathcal{H}_{M1,m}) = \emptyset \wedge \\ & \mathcal{R}_{ENV}(\mathcal{H}_{M2}) \cap \mathcal{R}_{ENV}(\mathcal{H}_{M1,m}) = \emptyset \big) \wedge \\ & \mathcal{R}_{ENV}(\mathcal{H}_{M2}) \cap \mathcal{R}_{ENV}(\mathcal{H}_{M0}) = \emptyset \end{aligned}$$

$$\forall m \forall n \left(m \neq n \wedge \mathcal{C}_m \in \mathcal{M} \cap \mathcal{U} \wedge \mathcal{C}_n \in \mathcal{M} \cap \mathcal{U} \Rightarrow \right. \quad (5.20)$$

$$\left. \mathcal{R}_{ENV}(\mathcal{H}_{M1,m}) \cap \mathcal{R}_{ENV}(\mathcal{H}_{M1,n}) = \emptyset \right)$$

The interpretation of the detectability condition 5.19 is straightforward when analyzed along with the detection requirement 5.17; it simply states that under no conditions two out of the three OR operands in the relation \mathcal{R}_{FTC-DI} are simultaneously true. It is worth noting that all elements pointed out in [29] are present in the above detectability definition: the fault dynamics is captured by the fault mode specified within the fault hypothesis, while the system dynamics and the disturbances are modeled within $\mathcal{R}_{ENV}()$.

Chapter 6

Conclusions

The outcome of this research work is the formal specification of the requirements for the FTC system. This system interfaces with the AFCS to provide fault tolerance capability with respect to sensor failures. The developed specification represents the baseline specification for the FTC system, the starting point for the refinement iteration that takes place within the system life-cycle. The FTC specification was formulated on top of the AFCS performance specification and of the DHC-2 detail specification. The fault tolerance requirements were extracted from the active military specification MIL-F-9490D and translated in the context of analytical redundancy after a detailed analysis of the implications of adopting this redundancy approach. The long process of analysis, modeling, and specification of the FTC requirements resulted in a clear definition of the fault tolerance problem in the framework of analytical redundancy. Furthermore, it brought up to light important issues in all fields involved in the process, namely FTFCS specification, analytical redundancy, and requirements engineering. These issues and the lessons learned are summarized below.

The re-engineering process of the active FCS specification uncovered the ambiguity and incompleteness of the FCS performance and fault tolerance requirements.

These potentially dangerous characteristics of the FCS specification might not represent a treat since in practice the objectives of FCS certification go way beyond the fulfillment of the *written* requirements; however, they do highlight the immaturity of the specification. Mature and formally specified requirements would result in a more effective design and a more efficient certification process eventually supported by automated tools.

A first observation about the analytical redundancy approach is that it can only be used to provide fault tolerance with respect to failure of components that are *functionally* redundant within the system. This implies that some degree of physical redundancy is required within the FTFCS.

Two important characteristics of analytical redundancy based solutions originate from these systems being subordinated to the operation of a considerable number of components of their environment. This dependence has an impact on the modularity and certifiability of the FTC system. If different FTC modules are to be adopted for different classes of components a priority ordering must be specified. This ordering determines a stratified architecture of the FTC system where for each FTC module, faults of components monitored at the lower layers are transparent, while faults of components monitored at the upper layers are not. Despite the modularity of the FTC architecture fault tolerance requirements for each FTC module must be total with respect to operation of all fallible components. The priority ordering and the totality characteristic of fault tolerance requirements diminish the typical advantages of the modular approach.

The FTC intrinsic dependence on the dynamics of the aircraft also raises a significant issue in the certification of such systems: they potentially require a certification process comparable to that adopted in the certification of AFCS's, with all related consequences in terms of costs, time, and employed resources. Furthermore, the fault modes of the aircraft and FTFCs components form the basis of the certification. This implies that the FTC must be certified against each combination of faulty components and fault-modes as captured by the fault hypotheses. The fault modes of all components whose failures are not transparent to the FTC module come to play a crucial role in the feasibility of the solution.

The above observations point out some significant implications of adopting the analytical redundancy approach as a basis for fault tolerance. The core message of these observations is that the FTC fault tolerance requirements and related certification procedures are considerably more demanding than those typically adopted in the literature. The more stringent requirements should not discourage the FTC designer, rather they should be regarded as the basis for a more rationale and effective design. Functional redundancy is a form of redundancy, it provides – to some degree – the means to achieve fault tolerance. The question is not whether analytical redundancy can be used, but how to integrate it with physical redundancy to effectively achieve fault tolerance. Adopting analytical redundancy to reduce the level of physical redundancy in multistring architecture from triple or quadruple to dual would bring considerable benefits already.

The observations related to requirements engineering converge upon the specifi-

cation methodology and the suitability of relational algebra as formal specification language. A number of issues related to the hierarchical decomposition of the specification, to the modeling of the specification, and to the dependencies among variables that are not part of the system input/output interface caused non-trivial problems. The lack of a mature methodology for developing the specification of a fairly complex system as the one under analysis in this research work considerably slowed down the specification process. Another important issue related to the specification process is the need of automatic tools to facilitate development and documentation. The complexity of the specification, the intrinsic interdependence among its parts, and the continuous need of modifications require a well engineered development environment. Without automatic development and management tools the use of a formal specification does not deliver all the benefits it could.

Among the most desirable characteristics of a formal specification language the author lists *monotonicity*. Monotonicity is a fundamental feature, it allows modifying and updating the specification over time, thus providing continuity and consistency during the development of the specification. If it is true – and it is – that the formalization process of requirements leads to a better understanding of the problem to address, then the requirements are continuously updated during this process. This explains why monotonicity is so important. The author’s experience with non-monotonic specification languages ended up with useless specifications. Monotonicity of relational algebra was one of the features that pointed toward its choice, to damage of other languages that come with support tools. Another advantage of relational algebra

is that predicate logic specifications are readable and do not require any unnatural translation into a rigid specification model.

The development of a formal specification requires a considerable amount of resources. On the other hand, the efforts employed at producing a formal specification pay off with a remarkable understanding of the problem under analysis and an unambiguous and upgradable formulation of the requirements. Formal specification is advisable whenever the system under development raises safety issues.

Bibliography

- [1] Background information and user guide for MIL-F-9490D. Technical report AFFDL-TR-74-116, USAF, 1975.
- [2] Flight control systems - design, installation and test of piloted aircraft, general specification for. Military specification MIL-F-9490D, USAF, 1975.
- [3] Appendix to background information and user guide for MIL-F-9490D. Technical report AFFDL-TR-74-116 sup1, USAF, 1980.
- [4] Flying qualities for piloted airplanes. Technical Report MIL-F-8785C, USAF, 1980.
- [5] Background information and user guide for MIL-F-8785C. Technical report AFWAL-TR-81-3109, USAF, 1982.
- [6] IEEE guide for developing system requirements specifications. Standard IEEE Std 1233, IEEE, 1998.
- [7] B. D. Appleby, J. R. Dowdle, and W. Vander Velde. Robust estimator design using μ synthesis. *Proc. of the 30th conference on Decision and Control*, pages 640–644, 1991.
- [8] M. Basseville. Detecting changes in signals and systems - a survey. *Automatica*, 24:309–326, 1988.
- [9] B. W. Boehm. Verifying and validating software requirements and design specifications. *IEEE Software*, 1:75–88, 1984.
- [10] F. W. Burcham, T. A. Maine, and C. Gordon. Development and flight test of an emergency flight control system using only engine thrust on an F-15 aircraft. Technical Report 94-02, NASA Dryden Flight Research Center, 1994.
- [11] J. Chen and R.J. Patton. *Robust Model-Based Fault Diagnosis for Dynamic Systems*. Kluwer Academic Publishers, 1999.
- [12] J. Chen and H. Y. Zhang. Parity vector approach for detecting failures in dynamic systems. *Int. Journal Sys. Sci.*, pages 765–770, 1990.

- [13] Jie Chen and R.J. Patton. A re-examination of fault detectability and isolability in linear dynamic systems. *Fault Detection, Supervision and Safety for Technical Processes*, pages 567–73, 1994.
- [14] E. Y. Chow and A. S. Willsky. Issues in the development of a general algorithm for reliable failure detection. *Proc. of the 19th Conf. on Decision and Control*, 1980.
- [15] E. Y. Chow and A. S. Willsky. Analytical redundancy and the design of robust detection systems. *IEEE Trans. on Automatic Control*, 29:603–614, 1984.
- [16] R. N. Clark. The dedicated observer approach to instrument failure detection. *Proc. of the 18th IEEE Conf. on Decision and Control*, pages 237–241, 1979.
- [17] J. C. Deckert, M. N. Desai, J. J. Deyst, and A. S. Willsky. F-8 DFBW sensor failure identification using analytical redundancy. *IEEE Transactions on Automatic Control*, 22:795–803, 1977.
- [18] X. Ding, L. Guo, and P. M. Frank. A frequency domain approach to fault detection of uncertain systems. *Proc. of the 32nd IEEE Conference on Decision and Control*, pages 1722–1727, 1993.
- [19] M. Dorfman. Requirements engineering. *Software Requirements Engineering*, pages 7–22, 1997.
- [20] Brooks F. No silver bullet: Essence and accidents of software engineering. *Computer*, April 1987.
- [21] C. Favre. Fly-by-wire for commercial aircraft: The airbus experience. *International Journal of Control*, 59:139–157, 1994.
- [22] P. M. Frank. Fault diagnosis in dynamic system via state estimation - a survey. in *Tzafestas, Singh, and Shmidt (Eds.), System Fault Diagnostics, Reliability and Related Knowledge-based Approaches*, D. Reidel Press, Dordrecht, pp.35-98, 1987.
- [23] P. M. Frank and J. Wunnenberg. Robust fault diagnosis using unknown input schemes. in *R. J. Patton, P. M. Frank, and R. N. Clark (Eds.), "Fault Diagnosis in Dynamic Systems: Theory and Application"*, Prentice Hall, chapter 3, pp.47-98, 1989.
- [24] P.M. Frank and B. Koppen. Review of optimal solutions to the robustness problem in observer-based fault detection. *Journal of Systems and Control Engineering*, 207(12):105–12, 1993.
- [25] J. Gertler. Fault detection and isolation using parity relations. *Control Eng. Practice*, 5:653–661, 1997.

- [26] J. Gertler and M. Costin. Model-based diagnosis of automotive engines-case study on a physical vehicle. *IFAC Symposium on Fault Detection, Supervision and Safety for Technical Processes - SAFEPROCESS '94*, 2:421–430, 1994.
- [27] J. Gertler and G. DiPierro. On the relationship between parity relations and parameter estimation. *Proc. of the IFAC Sympo. on Fault Detection, Supervision and Safety for Technical Processes: SAFEPROCESS'97*, pages 453–458, 1997.
- [28] J. Gertler, Q. Luo, K. Anderson, and X. W. Fang. Diagnosis of plant failures using orthogonal parity equations. *Proc. of the 11th IFAC World Congress*, 1990.
- [29] D. Del Gobbo and M. Napolitano. Issues in fault detectability for dynamic systems. *American Control Conference*, 2000.
- [30] D. M. Himmelblau. Fault detection in heat exchangers. *Proceedings of the 1992 American Control Conference*, 2:2369–2372, 1992.
- [31] D. E. Hoak, D. E. Ellison, and et al. *USAF Stability and Control DAT-COM*. Flight Control Division, Air Force Flight Dynamics Laboratory; Wright-Patterson Air Force Base, Ohio., 1968.
- [32] D.T. Horak. Failure detection in dynamic systems with modeling errors. *Journal of Guidance, Control, and Dynamics*, 11(6):508–16, 1988.
- [33] R. Isermann. Process fault detection based on modelling and estimation methods: A survey. *Automatica*, 20:387–404, 1984.
- [34] R. Isermann. Experiences with process fault detection via parameter estimation. in *Tzafestas, Singh, and Shmidt (Eds.), System Fault Diagnostics, Reliability and Related Knowledge-based Approaches*, D. Reidel Press, Dordrecht, pp. 3-33, 1987.
- [35] R. O. Lewis. *Independent Verification and Validation: A Life Cycle Engineering Process for Quality Software*. John Wiley & Sons, 1992.
- [36] Jackson M. The meaning of requirements. *Annals of Software Engineering*, 3:5–21, 1997.
- [37] Rauw M. FDC 1.2 - A Simulink toolbox for flight dynamics and control analysis - user manual. <http://www.mathworks.com/>, 1998.
- [38] Rauw M.O. A Simulink environment for flight dynamics and control analysis - application to the DHC-2 'beaver'. *Graduate's thesis. Delft Univ. of Thechnology, the Netherlands*, 1993.
- [39] M. R. Napolitano, V. Casdorph, C. Neppach, S. Naylor, M. Innocenti, and G. Silvestri. Online learning neural architectures and cross-correlation analysis for actuator failure detection and identification. *International Journal of Control*, 63:433–455, 1996.

- [40] M. R. Napolitano, C. I. Chen, and S. Naylor. Aircraft failure detection and identification using neural networks. *Journal of Guidance, Control, and Dynamics*, 16:999–1009, 1993.
- [41] M. R. Napolitano, D. A. Windon, J. L. Casanova, M. Innocenti, and G. Silvestri. Kalman filters and neural-network schemes for sensor validation in flight control systems. *IEEE Transactions on Control Systems Technology*, 6:596–611, 1998.
- [42] B. A. Nuseibeh and S. M. Easterbrook. Requirements engineering: A roadmap. *Proceedings, 22nd International Conference on Software Engineering (ICSE'00)*, pages 35–46, June 4–11 2000.
- [43] M. Nyberg and L. Nielsen. Parity functions as universal residual generators and tool for fault detectability analysis. *36th IEEE Conference on Decision and Control*, 5:4483–9, 1997.
- [44] S. Osder. Practical view of redundancy management-application and theory. *Journal of Guidance, Control, and Dynamics*, 22:12–21, 1999.
- [45] J. H. Park, Y. Halevi, and G. Rizzoni. A new interpretation of the fault-detection filter. 2: Theoptimal detection filter. *Int. Journal of Control*, 60:1339–1351, 1994.
- [46] J. H. Park and G. Rizzoni. A new interpretation of the fault-detection filter. 1: Closed-form algorithm. *Int. Journal of Control*, 60:767–787, 1994.
- [47] R. J. Patton. Robust model-based fault diagnosis: The state of the art. *IFAC Symposium on Fault Detection, Supervision and Safety for Technical Processes - SAFEPROCESS '94*, 1:1–24, 1994.
- [48] R. J. Patton and J. Chen. A re-examination of the relationship between parity space and observer-based approaches in fault diagnosis. *European Journal of Diagnosis and Safety in Automation*, 1:183–200, 1991.
- [49] R. J. Patton and J. Chen. Robust fault detection of jet engine sensor systems using eigenstructure assignment. *Journal of Guidance, Control and Dynamics*, 15:1491–1497, 1992.
- [50] R. J. Patton, J. Chen, and C. J. Lopez-Toribio. Fuzzy observers for non-linear dynamic systems fault diagnosis. *Proceedings of the 37th IEEE Conference on Decision and Control*, 1:84–89, 1998.
- [51] R.J. Patton, P. M. Frank, and R. N. Clark (Eds.). *Issues of Fault Diagnosis for Dynamic Systems*. Springer Verlag, 2000.
- [52] Patton R.J., Frank P., and Clark R. *Fault Diagnosis in Dynamic Systems, Theory and Applications*. Prentice Hall, 1989.

- [53] Tjee R.T.H. and Mulder J.A. Stability and control derivatives of the De Havilland DHC-2 'beaver' aircraft. *Report LR-556, Delft Univ. of Technology, The Netherlands*, 1988.
- [54] M. A. Sadrnia, J. Chen, and R. J. Patton. Robust fault detection observer design using H_∞/μ techniques for uncertain flight control systems. *Proc. of the 2ndIFAC Symposium on Robust Control Design: RECOND97*, pages 531–536, 1997.
- [55] G. Schmidt and T. Strohlein. *Relations and Graphs*. Springer Verlag, 1993.
- [56] K.J. Szalai, R.R. Larson, and R.D. Glover. Flight experience with flight control redundancy management. In *AGARD Lecture Series No.109. Fault Tolerance Design and Redundancy Management Techniques*, AGARD Lecture Series, pages 8/1–27, AGARD, Neuilly-sur-Seine, France, October 1980. AGARD.
- [57] A. T. Vemuri and M. M. Polycarpou. Neural-network-based robust fault diagnosis in robotic systems. *IEEE Transactions on Neural Networks*, 8:1410–1420, 1997.
- [58] Alagar V.S. and Periyasamy K. *Specification of Software Systems*. Springer Verlag, 1998.
- [59] J. L. Weiss and J. Y. Hsu. Design and evaluation of a failure detection and isolation algorithm for restructurable control systems. *Technical Report NASA-CR-178213*, 1985.
- [60] A. S. Willsky. A survey of design methods for failure detection in dynamic systems. *Automatica*, 12:601–611, 1976.
- [61] A. S. Willsky, J. J. Deyst, and B. S. Crawford. Adaptive filtering and self-test methods for failed detection and compensation. *Proc. of the 1974 Joint American Control Conf.*, pages 637–645, 1974.
- [62] Y.C. Yeh. Triple-triple redundant 777 primary flight computer. *IEEE Aerospace Applications Conference*, February 1996.

Appendix A

Predicate Logic and Relational Algebra

This appendix provides a formal definition of syntax and semantics of predicate logic and relational algebra. Predicate logic is the formal language adopted in relational specifications. Relational algebra is the mathematical framework that allows operating with relational specifications. The content of this appendix serves as support to chapters 4 and 5, and to appendix B. The presented information is largely based on references [55] and [58].

A.1 Logic

Logic can be used as a formal specification language. Examples of logic include *propositional logic* and *first-order predicate logic*. The information presented in this section provides a brief description of the language aspects of logic.

A.1.1 Propositional logic

A *proposition* is a statement that is either *True* (T) or *False* (F). *Propositional logic* consists of *sentences* constructed from *atomic formulas* and the *logical connectives*

Table A.1: Syntax of propositional logic

<i>terminals</i>	=	$\{P, Q, R, \dots, \neg, \wedge, \vee, \Rightarrow, \Leftrightarrow, (,);\}$
<i>nonterminals</i>	=	$\{atomic\ formula, sentence\};$
<i>atomic formula</i>	=	$P \mid Q \mid R \mid \dots;$
<i>sentence</i>	=	$atomic\ formula \mid (, sentence,) \mid \neg, sentence \mid$ $sentence, \wedge, sentence \mid$ $sentence, \vee, sentence \mid$ $sentence, \Rightarrow, sentence \mid$ $sentence, \Leftrightarrow, sentence;$

\wedge (*and*), \vee (*or*), \neg (*not*), \Rightarrow (*if ... then*), \Leftrightarrow (*if and only if*). Atomic formulas are the simplest form of propositions. Examples of atomic formulas are:

$$\begin{aligned} 3 &> 2 \\ a &> 5 \\ a &> b \end{aligned}$$

Examples of sentences are:

$$\begin{aligned} 3 &> 2 \wedge 2 > 1 \\ a &> 2 \Rightarrow b > 1 \\ a &> b \wedge b > 2 \Rightarrow c > 2 \end{aligned}$$

Table A.1 defines the syntax of the language by means of the Backus Naur formalism, while table A.2 defines its semantics. The logic operators are subject to the following precedence ordering: $\neg, \wedge, \vee, \Rightarrow, \Leftrightarrow$; \neg being the operator with highest precedence. The semantics of a sentence is obtained by assigning truth values (T or F) to atomic formulas and evaluating the sentences according to the semantics of the language.

A.1.2 Predicate logic

Table A.3 defines the syntax of predicate logic. Constants and connectives are interpreted as in propositional logic. Predicates are functions that evaluate either true

Table A.2: Semantics of propositional logic

P	Q	$\neg P$	$P \vee Q$	$P \wedge Q$	$P \Rightarrow Q$	$P \Leftrightarrow Q$
T	T	F	T	T	T	T
T	F	F	T	F	F	F
F	T	T	T	F	T	F
F	F	T	F	F	T	T

Table A.3: Syntax of predicate logic

$formula$	$=$	$proposition \mid predicate \mid \neg formula \mid$ $quantified_formula \mid$ $(, formula, op, formula,);$
$proposition$	$=$	$P \mid Q \mid R \dots;$
$predicate$	$=$	$predicate_name, (, term_list,);$
$predicate_name$	$=$	$IDENTIFIER$
$term_list$	$=$	$term \mid term, ", ", term_list;$
$term$	$=$	$CONSTANT \mid variable \mid$ $function, (, term_list,);$
$variable$	$=$	$VARNAME;$
$function$	$=$	$IDENTIFIER;$
$quantified_formula$	$=$	$quantifier, formula;$
$quantifier$	$=$	$\exists, variable \mid \forall, variable;$
op	$=$	$\wedge, \vee, \Rightarrow, \Leftrightarrow;$

or false. Variables are either quantified or *free*. Formulas in which every variable is quantified are called *closed* formulas. Closed formulas can be interpreted and evaluate either true or false. The *meaning* of a formula F is function of the free variables. Each assignment to the free variables within a specified domain D leads to an *interpretation* of the formula. The meaning of the formula is an assignment of a truth value for each possible interpretation. After the assignment of values to free variables the interpretation of a formula involves the following operations:

- evaluation of functions and predicates according to their semantics

- evaluation of propositions and non-quantified sub-formulas according to the semantics of propositional logic
- evaluation of quantified formulas according to the following semantics:
 - a quantified formula of the type $\forall x F$ evaluates true if F is true for every value of x in the domain of interpretation; otherwise it evaluates false
 - a quantified formula of the type $\exists x F$ evaluates true if there exists at least one value of x within the domain of interpretation for which F is true; otherwise it evaluates false

Each operation is carried out to reduce the arguments of the formula until the truth value of the formula can be determined.

A.2 Relational algebra and requirements specification

Predicate logic provides means to formally specify requirements. Relational algebra provides the formal framework for reasoning with requirement specifications. The following section introduces the main concepts of relational algebra, while the last section illustrates the main concepts of relational specification.

A.2.1 Basics of relational algebra

Given two spaces (or sets) \mathcal{A} and \mathcal{B} , a relation \mathcal{R} over $\mathcal{A} \times \mathcal{B}$ is a subset of $\mathcal{A} \times \mathcal{B}$ and is specified as follows:

$$\mathcal{R} = \{(\mathcal{A}, \mathcal{B}) \mid F(\mathcal{A}, \mathcal{B})\} \tag{A.1}$$

The above expression reads: *relation \mathcal{R} is the set of couples of elements $(\mathcal{A}, \mathcal{B}) \in \mathcal{A} \times \mathcal{B}$ such that $F(\mathcal{A}, \mathcal{B})$ evaluates true.* F is a predicate logic formula and is specified according to syntax and semantics rules presented in section A.1.2. In the following each relation is assumed to be over the space $\mathcal{A} \times \mathcal{B}$ unless otherwise specified.

Definitions

Relation domain and range If $(\mathcal{A}, \mathcal{B})$ is an element of relation \mathcal{R} , then \mathcal{A} is called an *antecedent* of \mathcal{R} and \mathcal{B} an *image* of \mathcal{R} . The set of images of \mathcal{A} by relation \mathcal{R} is denoted by $\mathcal{A} \bullet \mathcal{R}$, and the set of antecedents of element \mathcal{A} by relation \mathcal{R} is denoted by $\mathcal{R} \bullet \mathcal{A}$. The set of all antecedents of \mathcal{R} is the *domain* of relation \mathcal{R} and is denoted $dom(\mathcal{R})$. The set of all images of \mathcal{R} is the *range* (or *codomain*) of relation \mathcal{R} and is denoted $rng(\mathcal{R})$.

Universal relation The *universal* (or *total*) relation over the set \mathcal{A} is defined by

$\mathcal{A} \times \mathcal{A}$ and is denoted $\mathcal{L}_{\mathcal{A}}$.

Identity relation The *identity* relation over the set \mathcal{A} is defined by $\{(\mathcal{A}, \mathcal{A}') \mid \mathcal{A} = \mathcal{A}'\}$ and is denoted by $\mathcal{I}_{\mathcal{A}}$. Given a set $\mathcal{S} \subseteq \mathcal{A}$ we define $\mathcal{I}_{\mathcal{A}}(\mathcal{S}) = \{(\mathcal{A}, \mathcal{A}') \mid \mathcal{A} \in \mathcal{S} \wedge \mathcal{A} = \mathcal{A}'\}$.

Operations on relations

Since relations are sets, they inherit set operators: $\mathcal{P}(\mathcal{S})$ (*power set*), $\bar{\mathcal{S}}$ (*complement*), \times (*cartesian product*), \cap (*intersection*), \setminus (*difference*), \cup (*union*), and the inclusion ordering \supseteq . The precedence ordering of the set operators is the one adopted in introducing them; the power set operator being the one with highest

priority. Other operators more specific to relations are defined in the following.

Inverse The *inverse* of relation \mathcal{R} is the relation over $\subseteq \mathcal{B} \times \mathcal{A}$ denoted $\hat{\mathcal{R}}$ and defined by:

$$\hat{\mathcal{R}} = \{(\mathcal{B}, \mathcal{A}) \mid (\mathcal{A}, \mathcal{B}) \in \mathcal{R}\} \quad (\text{A.2})$$

Restriction The *prerestriction* of relation \mathcal{R} to subset \mathcal{S} of \mathcal{A} is the relation denoted by $\mathcal{S} \backslash \mathcal{R}$ and defined by:

$$\mathcal{S} \backslash \mathcal{R} = \mathcal{I}_{\mathcal{A}}(\mathcal{S}) \circ \mathcal{R} \quad (\text{A.3})$$

The *postrestriction* of relation \mathcal{R} to subset \mathcal{S} of \mathcal{B} is the relation denoted by $\mathcal{R} / \mathcal{S}$ and defined by:

$$\mathcal{R} / \mathcal{S} = \mathcal{R} \circ \mathcal{I}_{\mathcal{B}}(\mathcal{S}) \quad (\text{A.4})$$

A.2.2 Relational specifications

In terms of requirements specification expression A.1 is interpreted as follows: objects adopted in formulating the requirement are represented by *domain* and *image* variables; \mathcal{A} and \mathcal{B} are structures whose elements are the domain and image variables respectively; $F(\mathcal{A}, \mathcal{B})$ is a predicate logic formula that specifies the requirement in terms of domain and image variables, ad hoc introduced quantified variables, functions of such variables and constant values.

Relational algebra provides the formal framework for reasoning with predicate logic specifications. This framework is based on composition operators and a refinement ordering among requirement specifications. Composition operators allow

specification by parts. This specification method consists in breaking the required behaviour of the system into *parts*. Each *part* is specified by means of a relation. Then, these relations are composed together to form the whole requirements specification. The refinement ordering captures the idea of relative *strength* between two requirements. Also, it allows defining the *correctness* of a system implementation with respect to its requirements.

Definitions

Refinement ordering Relation \mathcal{R} is said to *refine* (or be a *refinement* of) relation

\mathcal{R}' (denoted by $\mathcal{R} \sqsupseteq \mathcal{R}'$) if and only if

$$\mathcal{R}\mathcal{L} \subseteq \mathcal{R}'\mathcal{L} \wedge \mathcal{R}'\mathcal{L} \cap \mathcal{R} \subseteq \mathcal{R}' \quad (\text{A.5})$$

$\mathcal{R} \sqsupseteq \mathcal{R}'$ implies

- $\text{dom}(\mathcal{R}) \supseteq \mathcal{R}'$, that is the requirement specified by \mathcal{R} extends over a larger (or equal) number of input scenarios, or
- $\forall \mathcal{A}(\mathcal{A} \in \text{dom}(\mathcal{R}') \Rightarrow \mathcal{A} \bullet \mathcal{R} \subseteq \mathcal{A} \bullet \mathcal{R}')$, that is relation \mathcal{R} is more specific in its assignment of outputs to inputs.

The defined ordering among specifications reflects the strength of the related requirements. Relation \mathcal{R} refines relation \mathcal{R}' if it defines a stronger (harder to satisfy) requirement. If $\mathcal{R} \sqsupseteq \mathcal{R}'$, then any system that satisfies \mathcal{R} satisfies \mathcal{R}' .

If $\mathcal{R} \sqsupseteq \mathcal{R}'$, then any system that satisfies \mathcal{R} satisfies \mathcal{R}' .

Correctness A system implementation \mathcal{P} is said to be correct with respect to its specification \mathcal{R} if and only if

$$\mathcal{P} \sqsupseteq \mathcal{R} \quad (\text{A.6})$$

Product The *product* of relation $\mathcal{R}_1 \subseteq \mathcal{A} \times \mathcal{K}$ by relation $\mathcal{R}_2 \subseteq \mathcal{K} \times \mathcal{B}$ is the relation over $\mathcal{A} \times \mathcal{B}$ denoted by $\mathcal{R}_1 \circ \mathcal{R}_2$ (or $\mathcal{R}_1 \mathcal{R}_2$) and defined by:

$$\mathcal{R}_1 \circ \mathcal{R}_2 = \{(\mathcal{A}, \mathcal{B}) \mid \exists \mathcal{K} ((\mathcal{A}, \mathcal{K}) \in \mathcal{R}_1 \wedge (\mathcal{K}, \mathcal{B}) \in \mathcal{R}_2)\} \quad (\text{A.7})$$

Join The sum of the requirement information of two relations $\mathcal{R}_1 \subseteq \mathcal{A} \times \mathcal{B}$ and $\mathcal{R}_2 \subseteq \mathcal{A} \times \mathcal{B}$ is called the *join* of \mathcal{R}_1 and \mathcal{R}_2 and is denoted by $\mathcal{R}_1 \sqcup \mathcal{R}_2$.

The join of two relations is defined as follows:

$$\mathcal{R}_1 \sqcup \mathcal{R}_2 = \mathcal{R}_1 \cap \overline{\mathcal{R}_2 \mathcal{L}} \cup \mathcal{R}_2 \cap \overline{\mathcal{R}_1 \mathcal{L}} \cup \mathcal{R}_1 \cap \mathcal{R}_2 \quad (\text{A.8})$$

The following implication holds:

$$\mathcal{R} \sqsupseteq \mathcal{R}_1 \sqcup \mathcal{R}_2 \Leftrightarrow \mathcal{R} \sqsupseteq \mathcal{R}_1 \wedge \mathcal{R} \sqsupseteq \mathcal{R}_2 \quad (\text{A.9})$$

The join exists if and only if \mathcal{R}_1 and \mathcal{R}_2 do not *contradict* each other. The *consistency* condition to check whether the join of relations \mathcal{R}_1 and \mathcal{R}_2 exists is the following:

$$\mathcal{R}_1 \mathcal{L} \cap \mathcal{R}_2 \mathcal{L} = (\mathcal{R}_1 \cap \mathcal{R}_2) \mathcal{L} \quad (\text{A.10})$$

Meet The common requirement information of two relations $\mathcal{R}_1 \subseteq \mathcal{A} \times \mathcal{B}$ and

$\mathcal{R}_2 \subseteq \mathcal{A} \times \mathcal{B}$ is called the *meet* of \mathcal{R}_1 and \mathcal{R}_2 and is denoted by $\mathcal{R}_1 \sqcap \mathcal{R}_2$.

The meet of two relations is defined as follows:

$$\mathcal{R}_1 \sqcap \mathcal{R}_2 = \mathcal{R}_1 \mathcal{L} \cap \mathcal{R}_2 \mathcal{L} \cap (\mathcal{R}_1 \cup \mathcal{R}_2) \quad (\text{A.11})$$

The following implication holds:

$$\mathcal{R} = \mathcal{R}_1 \sqcap \mathcal{R}_2 \Rightarrow \mathcal{R}_1 \supseteq \mathcal{R} \wedge \mathcal{R}_2 \supseteq \mathcal{R} \quad (\text{A.12})$$

Expansion operation If \mathcal{R} is a relation over the space $\mathcal{S} = \mathcal{A} \times \mathcal{B}$, its expansion

over the space $\mathcal{S}' = (\mathcal{A}, \mathcal{A}') \times (\mathcal{B}, \mathcal{B}')$ is defined as follows:

$$\sigma_S^{S'} \circ \mathcal{R} \circ \hat{\sigma}_S^{S'} \quad (\text{A.13})$$

where the operator $\sigma_S^{S'}$ is defined by:

$$\sigma_S^{S'} = \{ (\mathcal{S}, \mathcal{S}') \mid \mathcal{A}(\mathcal{S}) = \mathcal{A}(\mathcal{S}') \wedge \mathcal{B}(\mathcal{S}) = \mathcal{B}(\mathcal{S}') \} \quad (\text{A.14})$$

The expansion operation is used to expand the spaces over which the relation is defined in order to apply the composition operators.

In a relational specification operators from relational algebra, set theory, and logic may be found in the same expression (see eq. A.5). The precedence between this class of operators is the following: relational operators are evaluated first, then set operators, then logic operators. The precedence ordering between operators of the same class has been defined in section A.1.1 for the logic operators, and in section

A.2.1 for the set operators. The precedence ordering among relational operators is the following: restriction, inverse, complement, product, meet, join; restriction being the operator with higher priority.

Appendix B

Elementary specifications of the AR-FTC environment

This appendix collects the elementary specifications of the AR-FTC environment. Requirements are specified by means of relations according to the syntax and semantics introduced in appendix A. Elementary specifications are separated into five different sections. Section B.1 collects relations used in the AFCS performance specification, section B.2 collects relations used in the DHC-2 detail-specification, section B.3 collects relations describing fault-modes, section B.4 collects the restriction sets, and section B.5 collects relations describing the interface blocks of the FTC system to the AR-FTC module.

B.1 Elementary requirements of AFCS performance specification

3.1.2 AFCS performance requirements ... Unless otherwise specified, these requirements apply in smooth air and include sensor error. ...

3.1.2.1 Attitude Hold (pitch and roll)

- a) Attitudes $(\theta(), \phi()) \in I$ shall be maintained in smooth air ($turb() \in A$) with a static accuracy of ± 0.5 degree in pitch attitude ($\theta_{acc} \in C$)
- b) (with wings level) ($phi() \in D; \phi_{acc} \in C$)
- c) and ± 1.0 degree in roll attitude ($\phi_{acc} \in C$) with respect to the reference ($\theta_r(), \phi_r() \in D; constRef() \in A$).
- d) RMS ($RMS() \in A$) attitude deviations shall not exceed 5 degrees in pitch ($\theta_{RMS} \in C$)
- e) or 10 degrees in roll attitude ($\phi_{RMS} \in C$) in turbulence ($turb() \in A$) at the intensities specified in 3.1.3.7 ($u_{wt}(), v_{wt}(), w_{wt}(), u_{wg}(), v_{wg}(), w_{wg}() \in D$).
- f) Accuracy requirements shall be achieved and maintained within 5 seconds ($T_\theta, T_\phi \in C$) of mode engagement ($SW_{PAH}(), SW_{RAH}() \in D; engaged() \in A$)
- g) for a 5 degree attitude disturbance ($\Delta\theta^*, \Delta\phi^* \in C$).

Intermediate specification (PAH)

FOR EVERY couple of time instants t_1, t_2

IF

- $[t_1, t_2]$ is a time interval within $[0, \infty)$ AND
- f) the length of the time interval $[t_1, t_2]$ is larger than T_θ AND
- b) the aircraft is wings level throughout the interval $[t_1, t_2]$ AND
 - the PAH control function is engaged at $t = t_1$ and stays engaged throughout the interval $[t_1, t_2]$ AND
 - the reference pitch is constant throughout the interval $[t_1, t_2]$ AND
- g) the pitch deviation from the reference at engagement is $\Delta\theta^*$

THEN

- a) IF there is no turbulence within the time interval $[t_1, t_2]$ THEN the deviation of the pitch angle from the reference shall not be larger than θ_{acc} at all time instants within the time interval $[t_1 + T_\theta, t_2]$ AND

- d) IF there is turbulence within the time interval $[t_1, t_2]$ THEN the RMS value of the deviation of the pitch angle from the reference over the time interval $[t_1 + T_\theta, t_2]$ shall not be larger than θ_{RMS}

Relational specification (PAH)

$$\begin{aligned} \mathcal{R}_{PAH} = & \quad (B.1) \\ & \left\{ \left((\mathcal{U}_w, \mathcal{U}_p, \mathcal{X}), \mathcal{X}' \right) \middle| \forall t_1 \forall t_2 \left(0 \leq t_1 < t_2 < \infty \wedge t_2 > t_1 + T_\theta \wedge \right. \right. \\ & \quad \forall t \left(t_1 \leq t \leq t_2 \Rightarrow |\phi(t)| < \phi_{acc} \right) \wedge \\ & \quad engaged(SW_{PAH}(), t_1, t_2) \wedge \\ & \quad constRef(\theta_r(), t_1, t_2) \wedge \\ & \quad |\theta(t_1) - \theta_r(t_1)| = \Delta\theta^* \Rightarrow \\ & \quad \left(\neg turb(t_1, t_2) \Rightarrow \forall t \left(t_1 + T_\theta \leq t \leq t_2 \Rightarrow |\theta(t) - \theta_r(t_1)| < \theta_{acc} \right) \right) \wedge \\ & \quad \left(turb(t_1, t_2) \Rightarrow RMS(\theta() - \theta_r(t_1), t_1 + T_\theta, t_2) < \theta_{RMS} \right) \\ & \left. \right\} \end{aligned}$$

Intermediate specification (RAH)

FOR EVERY couple of time instants t_1, t_2

IF

- $[t_1, t_2]$ is a time interval within $[0, \infty)$ AND
- f) the length of the time interval $[t_1, t_2]$ is larger than T_θ AND
- the RAH control function is engaged at $t = t_1$ and stays engaged throughout the interval $[t_1, t_2]$ AND
- the reference bank is constant throughout the interval $[t_1, t_2]$ AND
- g) the roll deviation from the reference at engagement is $\Delta\phi^*$

THEN

- c) IF there is no turbulence within the time interval $[t_1, t_2]$ THEN the deviation of the bank angle from the reference shall not be larger than ϕ_{acc} at all time instants within the time interval $[t_1 + T_\phi, t_2]$ AND
- e) IF there is turbulence within the time interval $[t_1, t_2]$ THEN the RMS value of the deviation of the bank angle from the reference over the time interval $[t_1 + T_\phi, t_2]$ shall not be larger than ϕ_{RMS}

Relational specification (RAH)

$$\mathcal{R}_{RAH} = \left\{ \left((\mathcal{U}_w, \mathcal{U}_p, \mathcal{X}), \mathcal{X}' \right) \mid \forall t_1 \forall t_2 \left(0 \leq t_1 < t_2 < \infty \wedge t_2 > t_1 + T_\phi \wedge \right. \right. \\ \left. \left. \begin{aligned} & \text{engaged}(SW_{RAH}(), t_1, t_2) \wedge \\ & \text{constRef}(\phi_r(), t_1, t_2) \wedge \\ & |\phi(t_1) - \phi_r(t_1)| = \Delta\phi^* \Rightarrow \\ & \left(\neg \text{turb}(t_1, t_2) \Rightarrow \forall t \left(t_1 + T_\phi \leq t \leq t_2 \Rightarrow |\phi(t) - \phi_r(t_1)| < \phi_{acc} \right) \right) \wedge \\ & \left(\text{turb}(t_1, t_2) \Rightarrow \text{RMS}(\phi() - \phi_r(t_1), t_1 + T_\phi, t_2) < \phi_{RMS} \right) \end{aligned} \right. \right. \\ \left. \right\} \quad (\text{B.2})$$

3.1.2.2 Heading Hold

- a) In smooth air ($\text{turb}() \in A$), heading ($\psi() \in I$) shall be maintained within a static accuracy of ± 0.5 degree ($\psi_{acc} \in C$) with respect to the reference ($\psi_r \in Q$).
- b) In turbulence, RMS deviations ($\text{RMS}() \in A$; $u_{wt}(), v_{wt}(), w_{wt}(), u_{wg}(), v_{wg}(), w_{wg}() \in D$) shall not exceed 5 degrees ($\psi_{RMS} \in C$) in heading at the intensities specified in 3.1.3.7.
- c) When heading hold is engaged ($SW_{HH}() \in D$), the aircraft shall roll ($\phi() \in I$) towards wings level.
- d) The reference heading shall be that heading that exists when the aircraft passes through a roll attitude that is wings level plus or minus a tolerance ($\phi_{acc} \in C$).

Intermediate specification

FOR EVERY couple of time instants t_1, t_3

IF

$[t_1, t_3]$ is a time interval within $[0, \infty)$ AND

the HH control function is engaged at $t = t_1$ and stays engaged throughout the interval $[t_1, t_3]$

THEN

there must EXIST a time instant t_2 and a reference heading ψ_r such that

$c_1)$ t_2 is within the interval $[t_1, t_3)$ AND

$c_2)$ the bank angle is approximately zero within the time interval $[t_2, t_3)$ AND

- d) the reference heading ψ_r is the heading angle at $t = t_2$ AND
- a) IF there is no turbulence within the time interval $[t_1, t_3]$ THEN the deviation of the heading angle ψ from the referenced heading ψ_r shall not be larger than ψ_{acc} at all time instants within the time interval $[t_2, t_3]$ AND
- b) IF there is turbulence within the time interval $[t_1, t_3]$ THEN the RMS value of the deviation of the heading angle ψ from the referenced heading ψ_r over the time interval $[t_2, t_3]$ shall not be larger than ψ_{RMS}

Relational specification

$$\begin{aligned}
 \mathcal{R}_{HH} = & \tag{B.3} \\
 & \left\{ \left((\mathcal{U}_w, \mathcal{U}_p), \mathcal{X} \right) \mid \forall t_1 \forall t_3 \left(\right. \\
 & \quad 0 \leq t_1 < t_3 < \infty \wedge \\
 & \quad engaged(SW_{HH}(), t_1, t_3) \Rightarrow \\
 & \quad \exists t_2 \exists \psi_r \left(\right. \\
 & \quad \quad t_1 \leq t_2 < t_3 \wedge \\
 & \quad \quad \forall t \left(t_2 \leq t \leq t_3 \Rightarrow |\phi(t)| < \phi_{acc} \right) \wedge \\
 & \quad \quad \psi_r = \psi(t_2) \wedge \\
 & \quad \quad \neg turb(t_1, t_3) \Rightarrow \forall t \left(t_2 \leq t \leq t_3 \Rightarrow |\psi(t) - \psi_r| < \psi_{acc} \right) \wedge \\
 & \quad \quad turb(t_1, t_3) \Rightarrow RMS(\psi() - \psi_r, t_2, t_3) < \psi_{RMS} \left. \right) \\
 & \left. \right\}
 \end{aligned}$$

3.1.2.3 Heading Select

- a) The aircraft shall automatically turn ($\phi() \in I$) through the smallest angle
- b) to any heading ($\psi_r() \in D; constRef() \in A$) selected or preselected by the pilot and
- c) maintain that heading ($\psi() \in I$) to the tolerances ($\psi_{acc}, \psi_{RMS} \in C; turb(), RMS() \in A; u_{wt}(), v_{wt}(), w_{wt}(), u_{wg}(), v_{wg}(), w_{wg}() \in D$) specified for heading hold.
- d) The contractor shall determine a bank angle limit ($\phi^-, \phi^+ \in C$) which provides a satisfactory turn rate and precludes impending stall.
- e) The aircraft shall not overshoot the selected heading by more than 1.5 degrees ($\psi_{os} \in C$).

- f) Entry into and exit from the turn shall be smooth and rapid.
- g) The roll rate ($p() \in I$) shall not exceed 10 deg/sec ($p^+ \in C$) and
- h) roll acceleration shall not exceed 5 deg/sec/sec ($\dot{p}^+ \in C$).

Intermediate specification

FOR EVERY couple of time instants t_1, t_4

IF

$[t_1, t_4]$ is a time interval within $[0, \infty)$ AND

the HS control function is engaged at $t = t_1$ and stays engaged throughout the interval $[t_1, t_4]$ AND

the reference heading is constant throughout the interval $[t_1, t_4]$

THEN

- a) the heading deviation from the reference shall never exceed 180 degrees throughout the interval $[t_1, t_4]$ AND
- g) The roll rate shall not exceed p^+ throughout the interval $[t_1, t_4]$ AND
- h) roll acceleration shall not exceed \dot{p}^+ throughout the interval $[t_1, t_4]$ AND
- d) bank angle shall be in the interval $[\phi^-, \phi^+]$ throughout the interval $[t_1, t_4]$ AND
- e) there EXIST a time instant $t_2 \in [t_1, t_4]$ when the heading is close to the reference and it was not close at all time instants within the time interval $[t_1, t_2]$ and it will never get further than ψ_{os} at all time instants within the time interval $[t_2, t_4]$
- b), c) there EXIST a time instant $t_3 \in [t_2, t_4]$ such that IF there is no turbulence within the time interval $[t_1, t_4]$ THEN the deviation of the heading angle from the referenced heading shall not be larger than ψ_{acc} at all time instants within the time interval $[t_3, t_4]$ AND IF there is turbulence within the time interval $[t_1, t_4]$ THEN the RMS value of the deviation of the heading angle from the referenced heading over the time interval $[t_3, t_4]$ shall not be larger than ψ_{RMS}

Relational specification

$$\mathcal{R}_{HS} = \tag{B.4}$$

$$\left\{ \left((\mathcal{U}_w, \mathcal{U}_p), \mathcal{X} \right) \mid \forall t_1 \forall t_4 \left(0 \leq t_1 < t_4 < \infty \wedge \right. \right. \\ \left. \left. engaged(SW_{HS}(), t_1, t_4) \wedge constRef(\psi_r(), t_1, t_4) \Rightarrow \right. \right.$$

$$\begin{aligned}
& \forall t \left(t_1 \leq t \leq t_4 \Rightarrow \right. \\
& \quad |\psi(t) - \psi_r(t_1)| \leq \pi \wedge \phi^- < \phi(t) < \phi^+ \wedge \\
& \quad p(t) < p^+ \wedge \dot{p}(t) < \dot{p}^+ \left. \right) \wedge \\
& \exists t_2 \left(t_1 \leq t_2 \leq t_4 \wedge |\psi(t_2) - \psi_r(t_1)| < \psi_{acc} \wedge \right. \\
& \quad \forall t \left(t_1 \leq t \leq t_2 \Rightarrow |\psi(t) - \psi_r(t_1)| > \psi_{acc} \right) \wedge \\
& \quad \forall t \left(t_2 \leq t \leq t_4 \Rightarrow |\psi(t) - \psi_r(t_1)| < \psi_{os} \right) \left. \right) \\
& \exists t_3 \left(t_2 \leq t_3 \leq t_4 \wedge \right. \\
& \quad \neg \text{turb}(t_1, t_4) \Rightarrow \forall t \left(t_3 \leq t \leq t_4 \Rightarrow |\psi(t) - \psi_r(t_1)| < \psi_{acc} \right) \wedge \\
& \quad \text{turb}(t_1, t_4) \Rightarrow \text{RMS}(\psi() - \psi_r(t_1)r, t_3, t_4) < \psi_{RMS} \left. \right) \\
& \left. \right\}
\end{aligned}$$

3.1.2.4 Lateral acceleration and sideslip limits ... the following performance shall be provided whenever any lateral-directional AFCS function is engaged. Lateral acceleration refers to apparent (measured, sensed) body axis acceleration at the aircraft center of gravity.

3.1.2.4.1 Coordination in steady banked turns

- a) The incremental sideslip angle ($\beta() \in I$) shall not exceed 2 degrees ($\Delta\beta_{/SBT}^+ \in C$) from the trimmed value ($\beta_{tr} \in D$),
- b) and lateral acceleration ($A_{ycg}() \in I$) shall not exceed 0.03g ($A_{ycg/SBT}^+ \in C$),
- c) while at steady bank angle ($\phi_{ss} \in Q, \phi_{acc} \in C; \phi() \in D$) up to the maneuver bank angle limit ($\phi^-, \phi^+ \in C$) reached during normal maneuvers
- d) with the AFCS engaged ($SW_{RAH}(), SW_{HH}(), SW_{HS}() \in D; \text{whileEngaged}() \in A$).

Intermediate specification

FOR EVERY couple of time instants t_1, t_2

IF

- $[t_1, t_2]$ is a time interval within $[0, \infty)$ AND
- d) either RAH, HH, or HS control function is in engaged status throughout the interval $[t_1, t_2]$ AND
- c) the airplane is in a steady bank turn throughout the interval $[t_1, t_2]$ and the steady bank angle is within allowed limits AND

- there is no turbulence throughout the interval $[t_1, t_2]$

THEN

- the sideslip angle deviation from trim value shall not exceed $\Delta\beta_{/SBT}^+$ throughout the interval $[t_1, t_2]$ AND
- The lateral acceleration shall not exceed $A_{ycg/SBT}^+$ throughout the interval $[t_1, t_2]$

Relational specification

$$\begin{aligned} \mathcal{R}_{SBT} = & \quad (B.5) \\ & \left\{ \left((\mathcal{U}_w, \mathcal{U}_p, \mathcal{X}, \mathcal{C}_{tr}), \mathcal{X}' \right) \middle| \forall t_1 \forall t_2 \left(0 \leq t_1 < t_2 < \infty \wedge \right. \right. \\ & \quad \left(whileEngaged(SW_{RAH}(), t_1, t_2) \vee \right. \\ & \quad whileEngaged(SW_{HH}(), t_1, t_2) \vee \\ & \quad \left. whileEngaged(SW_{HS}(), t_1, t_2) \right) \wedge \\ & \quad \exists \phi_{ss} \left(\forall t \left(t_1 \leq t \leq t_2 \Rightarrow |\phi(t) - \phi_{ss}| < \phi_{acc} \right) \wedge |\phi_{ss}| > \phi_{acc} \right) \wedge \\ & \quad \neg turb(t_1, t_2) \Rightarrow \\ & \quad \forall t \left(t_1 \leq t \leq t_2 \Rightarrow |\beta(t) - \beta_{tr}| < \Delta\beta_{/SBT}^+ \wedge |A_{ycg}(t)| < A_{ycg/SBT}^+ \right) \\ & \left. \right\} \end{aligned}$$

3.1.2.4.3 Coordination in straight and level flight

- The accuracy while the aircraft is in straight and level flight ($\phi() \in D$)
- shall be maintained with an incremental sideslip angle ($\beta() \in I$) of ± 1 degree ($\Delta\beta_{/LF}^+ \in C$) from the trimmed value ($\beta_{tr} \in D$) or a lateral acceleration ($A_{ycg}() \in I$) of ± 0.02 g ($A_{ycg/LF}^+ \in C$) at the cg,
- whichever is lower.

Intermediate specification

FOR EVERY couple of time instants t_1, t_2

IF

- $[t_1, t_2]$ is a time interval within $[0, \infty)$ AND
- d) either RAH, HH, or HS control function is in engaged status throughout the interval $[t_1, t_2]$ AND
- c) the airplane is wings level throughout the interval $[t_1, t_2]$ AND

- there is no turbulence throughout the interval $[t_1, t_2]$

THEN

- the sideslip angle deviation from trim value shall not exceed $\Delta\beta_{/LF}^+$ throughout the interval $[t_1, t_2]$ AND
- The lateral acceleration shall not exceed $A_{ycg/LF}^+$ throughout the interval $[t_1, t_2]$

Relational specification

$$\begin{aligned} \mathcal{R}_{LF} = & \quad (B.6) \\ & \left\{ \left((\mathcal{U}_w, \mathcal{U}_p, \mathcal{X}, \mathcal{C}_{tr}), \mathcal{X}' \right) \mid \forall t_1 \forall t_2 \left(0 \leq t_1 < t_2 < \infty \wedge \right. \right. \\ & \quad \left(whileEngaged(SW_{RAH}(), t_1, t_2) \vee \right. \\ & \quad whileEngaged(SW_{HH}(), t_1, t_2) \vee \\ & \quad \left. whileEngaged(SW_{HS}(), t_1, t_2) \right) \wedge \\ & \quad \forall t \left(t_1 \leq t \leq t_2 \Rightarrow |\phi(t)| < \phi_{acc} \right) \wedge \\ & \quad \neg turb(t_1, t_2) \Rightarrow \\ & \quad \forall t \left(t_1 \leq t \leq t_2 \Rightarrow |\beta(t) - \beta(0)| < \Delta\beta_{/LF}^+ \wedge |A_{ycg}(t)| < A_{ycg/LF}^+ \right) \\ & \left. \right\} \end{aligned}$$

3.1.2.5 Altitude hold

- Engagement ($t_1 \in Q$) of the altitude hold function ($SW_{ALH}() \in D, engaged() \in A$) at rates of climb or descent ($\dot{H}(t_1) \in D$) less than 2000 fpm ($\dot{H}_{/ALH} \in C$)
- shall select the existing indicated barometric altitude ($H(t_1) \in D$) and control ($H() \in I$) the aircraft to this altitude as a reference.
- The resulting normal acceleration ($A_n() \in I$) shall not exceed 0.2g ($\Delta A_{n/LF}^+ \in C$) incremental ($A_n(0) \in D$).
- For engagement at rates above 2000 feet per minute the AFCS shall not cause any unsafe maneuvers.
- Within the aircraft thrust-drag capability and
- at steady bank angles ($\phi() \in D; \phi_{acc} \in C; \phi_{ss} \in Q$), the mode shall provide control accuracies ($H_{acc1}, H_{acc2}, H_{acc3}, H_{acc2\%}, H_{acc3\%} \in C$) shown in Table B.1 ($\phi_{1/ALH}, \phi_{2/ALH}, \phi_{3/ALH} \in C$).
- These accuracy requirements apply for airspeeds ($V_a() \in D$) up to Mach 1.0 ($V_{a/LF}^+$).

- h) Following engagement or perturbation of this mode at 2000 feet per minute or less, the specified accuracy shall be achieved within 30 seconds ($T_H \in C$).

Table B.1: Minimum acceptable control accuracy for ALH function

Bank angle (deg.)	0 - 1	1 - 30	30 - 60
Alt. (ft.) 0 to 30000	$\pm 30\text{ft.}$	$\pm 60\text{ ft or } \pm 0.3\%$ whichever is larger	$\pm 90\text{ ft or } \pm 0.4\%$ whichever is larger

Intermediate specification

FOR EVERY couple of time instants t_1, t_2

IF

- $[t_1, t_2]$ is a time interval within $[0, \infty)$ AND
- the length of the time interval $[t_1, t_2]$ is larger than T_H AND
- a) the ALH control function is engaged at $t = t_1$ and stays engaged throughout the interval $[t_1, t_2]$ AND
- a) the rate of climb or descent at $t = t_1$ is less than $\dot{H}_{/ALH}$ AND
 - the airplane is at a steady bank angle throughout the interval $[t_1, t_2]$ AND
- g) the airspeed is below $V_{a/LF}^+$ AND
 - there is no turbulence throughout the interval $[t_1, t_2]$

THEN

- c) the normal acceleration deviation from trim value shall not exceed $\Delta A_{n/LF}^+$ throughout the interval $[t_1, t_2]$ AND
- b) f) h) throughout the interval $[t_1 + T_H, t_2]$ the airplane steady state bank angle is either less than $\phi_{1/ALH}$ – in which case altitude deviation from $H(t_1)$ shall be less than H_{acc1} – OR the airplane steady state bank angle is within the interval $[\phi_{1/ALH}, \phi_{2/ALH}]$ – in which case altitude deviation from $H(t_1)$ shall be less than the maximum between H_{acc2} and $H_{acc2\%} \cdot H(t_1)$ – OR the airplane steady state bank angle is within the interval $[\phi_{2/ALH}, \phi_{3/ALH}]$ – in which case altitude deviation from $H(t_1)$ shall be less than the maximum between H_{acc3} and $H_{acc3\%} \cdot H(t_1)$ –.

Relational specification

$$\begin{aligned}
\mathcal{R}_{ALH} = & \tag{B.7} \\
& \left\{ \left((\mathcal{U}_w, \mathcal{U}_p, \mathcal{X}), \mathcal{X}' \right) \middle| \forall t_1 \forall t_2 \exists \phi_{ss} \left(0 \leq t_1 < t_2 < \infty \wedge t_2 > t_1 + T_H \wedge \right. \right. \\
& \quad engaged(SW_{ALH}(), t_1, t_2) \wedge \\
& \quad |\dot{H}(t_1)| < \dot{H}_{/ALH}^+ \wedge \\
& \quad \forall t \left(t_1 \leq t \leq t_2 \Rightarrow |\phi(t) - \phi_{ss}| < \phi_{acc} \wedge \right. \\
& \quad \quad \left. V_a(t) < V_{a/LF}^+ \right) \wedge \\
& \quad \neg turb(t_1, t_2) \Rightarrow \\
& \quad \forall t \left(t_1 \leq t \leq t_2 \Rightarrow A_n(t) < \Delta A_{n/LF}^+ \right) \wedge \\
& \quad \forall t \left(t_1 + T_H \leq t \leq t_2 \Rightarrow \right. \\
& \quad \quad \left(|\phi_{ss}| < \phi_{1/ALH} \wedge |H(t) - H(t_1)| < H_{acc1} \right) \vee \\
& \quad \quad \left(\phi_{1/ALH} \leq |\phi_{ss}| < \phi_{2/ALH} \wedge \right. \\
& \quad \quad \left. |H(t) - H(t_1)| < max(H_{acc2}, H_{acc2\%} \cdot H(t_1)) \right) \vee \\
& \quad \quad \left(\phi_{2/ALH} \leq |\phi_{ss}| < \phi_{3/ALH} \wedge \right. \\
& \quad \quad \left. |H(t) - H(t_1)| < max(H_{acc3}, H_{acc3\%} \cdot H(t_1)) \right) \\
& \quad \left. \right) \Big\}
\end{aligned}$$

B.2 Elementary requirements of DHC-2 detail-specification

B.2.1 DHC-2 airplane dynamics

Force equation requirement

$$\begin{aligned} \mathcal{R}_{Feq} = & \left\{ \left((C_f, \mathcal{X}), \mathcal{X}' \right) \mid \forall t \exists m() \left(\right. \\ & 0 \leq t < \infty \wedge m^- \leq m(t) \leq m^+ \Rightarrow \\ & \dot{V}_a(t) = \frac{1}{m(t)} (F_x(t) \cos \alpha(t) \cos \beta(t) + F_y(t) \sin \beta(t) + \\ & \quad + F_z(t) \sin \alpha(t) \sin \beta(t)) \wedge \\ & \dot{\alpha}(t) = \frac{1}{m(t)V_a(t) \cos \beta(t)} (-F_x(t) \sin \alpha(t) + F_z(t) \cos \alpha(t)) + \\ & \quad + q(t) - (p(t) \cos \alpha(t) + r(t) \sin \alpha(t)) \tan \beta(t) \wedge \\ & \dot{\beta}(t) = \frac{1}{m(t)V_a(t)} (-F_x(t) \cos \alpha(t) \sin \beta(t) + F_y(t) \cos \beta(t) - \\ & \quad - F_z(t) \sin \alpha(t) \sin \beta(t)) + p(t) \sin \alpha(t) - r(t) \cos \alpha(t) \left. \right\} \end{aligned} \quad (B.8)$$

Airframe-exerted aerodynamic force requirement

$$\begin{aligned} \mathcal{R}_{aef} = & \left\{ \left(\mathcal{X}, C_f \right) \mid \forall t \exists m() \exists C_{X_0}() \exists C_{X_\alpha}() \exists C_{X_{\alpha^2}}() \exists C_{X_{\alpha^3}}() \right. \\ & \exists C_{X_q}() \exists C_{Y_0}() \exists C_{Y_\beta}() \exists C_{Y_p}() \exists C_{Y_r}() \exists C_{Y_{\dot{\beta}}}() \\ & \exists C_{Z_0}() \exists C_{Z_\alpha}() \exists C_{Z_{\alpha^3}}() \exists C_{Z_q}() \left(\right. \\ & 0 \leq t < \infty \wedge \\ & C_{X_0}(1 - \Delta C_{X_0\%}) \leq C_{X_0}(t) \leq C_{X_0}(1 + \Delta C_{X_0\%}) \wedge \\ & C_{X_\alpha}(1 - \Delta C_{X_\alpha\%}) \leq C_{X_\alpha}(t) \leq C_{X_\alpha}(1 + \Delta C_{X_\alpha\%}) \wedge \\ & C_{X_{\alpha^2}}(1 - \Delta C_{X_{\alpha^2}\%}) \leq C_{X_{\alpha^2}}(t) \leq C_{X_{\alpha^2}}(1 + \Delta C_{X_{\alpha^2}\%}) \wedge \\ & C_{X_{\alpha^3}}(1 - \Delta C_{X_{\alpha^3}\%}) \leq C_{X_{\alpha^3}}(t) \leq C_{X_{\alpha^3}}(1 + \Delta C_{X_{\alpha^3}\%}) \wedge \\ & C_{X_q}(1 - \Delta C_{X_q\%}) \leq C_{X_q}(t) \leq C_{X_q}(1 + \Delta C_{X_q\%}) \wedge \\ & C_{Y_0}(1 - \Delta C_{Y_0\%}) \leq C_{Y_0}(t) \leq C_{Y_0}(1 + \Delta C_{Y_0\%}) \wedge \\ & C_{Y_\beta}(1 - \Delta C_{Y_\beta\%}) \leq C_{Y_\beta}(t) \leq C_{Y_\beta}(1 + \Delta C_{Y_\beta\%}) \wedge \\ & C_{Y_p}(1 - \Delta C_{Y_p\%}) \leq C_{Y_p}(t) \leq C_{Y_p}(1 + \Delta C_{Y_p\%}) \wedge \end{aligned} \quad (B.9)$$

$$\begin{aligned}
C_{Y_r}(1 - \Delta C_{Y_r}\%) &\leq C_{Y_r}(t) \leq C_{Y_r}(1 + \Delta C_{Y_r}\%) \wedge \\
C_{Y_{\dot{\beta}}}(1 - \Delta C_{Y_{\dot{\beta}}}\%) &\leq C_{Y_{\dot{\beta}}}(t) \leq C_{Y_{\dot{\beta}}}(1 + \Delta C_{Y_{\dot{\beta}}}\%) \wedge \\
C_{Z_0}(1 - \Delta C_{Z_0}\%) &\leq C_{Z_0}(t) \leq C_{Z_0}(1 + \Delta C_{Z_0}\%) \wedge \\
C_{Z_{\alpha}}(1 - \Delta C_{Z_{\alpha}}\%) &\leq C_{Z_{\alpha}}(t) \leq C_{Z_{\alpha}}(1 + \Delta C_{Z_{\alpha}}\%) \wedge \\
C_{Z_{\alpha^3}}(1 - \Delta C_{Z_{\alpha^3}}\%) &\leq C_{Z_{\alpha^3}}(t) \leq C_{Z_{\alpha^3}}(1 + \Delta C_{Z_{\alpha^3}}\%) \wedge \\
C_{Z_q}(1 - \Delta C_{Z_q}\%) &\leq C_{Z_q}(t) \leq C_{Z_q}(1 + \Delta C_{Z_q}\%) \Rightarrow \\
X_{a_{sd}}(t) &= q_{dyn}(t)S \left(C_{X_0}(t) + C_{X_{\alpha}}(t)\alpha(t) + C_{X_{\alpha^2}}(t)\alpha^2(t) + \right. \\
&\quad \left. + C_{X_{\alpha^3}}(t)\alpha^3(t) + C_{X_q}(t)\frac{\bar{c}q(t)}{V_a(t)} \right) \wedge \\
Y_{a_{sd}}(t) &= q_{dyn}(t)S \left(C_{Y_0}(t) + C_{Y_{\beta}}(t)\beta(t) + C_{Y_p}(t)\frac{bp(t)}{2V_a(t)} + \right. \\
&\quad \left. + C_{Y_r}(t)\frac{br(t)}{2V_a(t)} + C_{Y_{\dot{\beta}}}(t)\frac{b\dot{\beta}(t)}{2V_a(t)} \right) \wedge \\
Z_{a_{sd}}(t) &= q_{dyn}(t)S \left(C_{Z_0}(t) + C_{Z_{\alpha}}(t)\alpha(t) + C_{Z_{\alpha^3}}(t)\alpha^3(t) + \right. \\
&\quad \left. + C_{Z_q}(t)\frac{\bar{c}q(t)}{V_a(t)} \right) \Bigg\}
\end{aligned}$$

Moment equation requirement

$$\begin{aligned}
\mathcal{R}_{Meq} &= \tag{B.10} \\
&\left\{ \left(\mathcal{C}_m, \mathcal{X} \right) \middle| \forall t \exists I_x() \exists I_y() \exists I_z() \exists J_{xy}() \exists J_{xz}() \exists J_{yz}() \left(\right. \right. \\
&\quad 0 \leq t < \infty \wedge \\
&\quad I_x(1 - \Delta I_x\%) \leq I_x(t) \leq I_x \wedge \\
&\quad I_y(1 - \Delta I_y\%) \leq I_y(t) \leq I_y \wedge \\
&\quad I_z(1 - \Delta I_z\%) \leq I_z(t) \leq I_z \wedge \\
&\quad J_{xy}(1 - \Delta J_{xy}\%) \leq J_{xy}(t) \leq J_{xy} \wedge \\
&\quad J_{xz}(1 - \Delta J_{xz}\%) \leq J_{xz}(t) \leq J_{xz} \wedge \\
&\quad J_{yz}(1 - \Delta J_{yz}\%) \leq J_{yz}(t) \leq J_{yz} \Rightarrow \\
&\quad \dot{p}(t) = P_{pp}(t)p^2(t) + P_{pq}(t)p(t)q(t) + P_{pr}(t)p(t)r(t) + P_{qq}(t)q^2(t) + \\
&\quad + P_{qr}(t)q(t)r(t) + P_{rr}(t)r^2(t) + P_l(t)L(t) + P_m(t)M(t) + P_n(t)N(t) \wedge \\
&\quad \dot{q}(t) = Q_{pp}(t)p^2(t) + Q_{pq}(t)p(t)q(t) + Q_{pr}(t)p(t)r(t) + Q_{qq}(t)q^2(t) + \\
&\quad + Q_{qr}(t)q(t)r(t) + Q_{rr}(t)r^2(t) + Q_l(t)L(t) + Q_m(t)M(t) + Q_n(t)N(t) \wedge \\
&\quad \dot{r}(t) = R_{pp}(t)p^2(t) + R_{pq}(t)p(t)q(t) + R_{pr}(t)p(t)r(t) + R_{qq}(t)q^2(t) + \\
&\quad + R_{qr}(t)q(t)r(t) + R_{rr}(t)r^2(t) + R_l(t)L(t) + R_m(t)M(t) + R_n(t)N(t) \Bigg\}
\end{aligned}$$

Airframe-exerted aerodynamic moment requirement

$$\mathcal{R}_{aem} = \quad (B.11)$$

$$\left\{ \left(\mathcal{X}, \mathcal{C}_m \right) \mid \forall t \exists m() \exists C_{l_0}() \exists C_{l_\beta}() \exists C_{l_p}() \exists C_{l_r}() \exists C_{m_\alpha}() \exists C_{m_{\alpha^2}}() \exists C_{m_q}() \right. \\
\left. \exists C_{m_{\beta^2}}() \exists C_{m_r}() \exists C_{n_0}() \exists C_{n_{\beta^3}}() \exists C_{n_\beta}() \exists C_{n_p}() \exists C_{n_r}() \exists C_{n_q}() \left(\right. \right. \\
0 \leq t < \infty \wedge \\
C_{l_0}(1 - \Delta C_{l_0}\%) \leq C_{l_0}(t) \leq C_{l_0}(1 + \Delta C_{l_0}\%) \wedge \\
C_{l_\beta}(1 - \Delta C_{l_\beta}\%) \leq C_{l_\beta}(t) \leq C_{l_\beta}(1 + \Delta C_{l_\beta}\%) \wedge \\
C_{l_p}(1 - \Delta C_{l_p}\%) \leq C_{l_p}(t) \leq C_{l_p}(1 + \Delta C_{l_p}\%) \wedge \\
C_{l_r}(1 - \Delta C_{l_r}\%) \leq C_{l_r}(t) \leq C_{l_r}(1 + \Delta C_{l_r}\%) \wedge \\
C_{m_0}(1 - \Delta C_{m_0}\%) \leq C_{m_0}(t) \leq C_{m_0}(1 + \Delta C_{m_0}\%) \wedge \\
C_{m_\alpha^-} \leq C_{m_\alpha}(t) \leq C_{m_\alpha^+} \wedge \\
C_{m_{\alpha^2}}(1 - \Delta C_{m_{\alpha^2}}\%) \leq C_{m_{\alpha^2}}(t) \leq C_{m_{\alpha^2}}(1 + \Delta C_{m_{\alpha^2}}\%) \wedge \\
C_{m_q}(1 - \Delta C_{m_q}\%) \leq C_{m_q}(t) \leq C_{m_q}(1 + \Delta C_{m_q}\%) \wedge \\
C_{m_{\beta^2}}(1 - \Delta C_{m_{\beta^2}}\%) \leq C_{m_{\beta^2}}(t) \leq C_{m_{\beta^2}}(1 + \Delta C_{m_{\beta^2}}\%) \wedge \\
C_{m_r}(1 - \Delta C_{m_r}\%) \leq C_{m_r}(t) \leq C_{m_r}(1 + \Delta C_{m_r}\%) \wedge \\
C_{n_0}(1 - \Delta C_{n_0}\%) \leq C_{n_0}(t) \leq C_{n_0}(1 + \Delta C_{n_0}\%) \wedge \\
C_{n_\beta}(1 - \Delta C_{n_\beta}\%) \leq C_{n_\beta}(t) \leq C_{n_\beta}(1 + \Delta C_{n_\beta}\%) \wedge \\
C_{n_p}(1 - \Delta C_{n_p}\%) \leq C_{n_p}(t) \leq C_{n_p}(1 + \Delta C_{n_p}\%) \wedge \\
C_{n_r}(1 - \Delta C_{n_r}\%) \leq C_{n_r}(t) \leq C_{n_r}(1 + \Delta C_{n_r}\%) \wedge \\
C_{n_q}(1 - \Delta C_{n_q}\%) \leq C_{n_q}(t) \leq C_{n_q}(1 + \Delta C_{n_q}\%) \wedge \\
C_{n_{\beta^3}}(1 - \Delta C_{n_{\beta^3}}\%) \leq C_{n_{\beta^3}}(t) \leq C_{n_{\beta^3}}(1 + \Delta C_{n_{\beta^3}}\%) \Rightarrow \\
L_{a_{sd}}(t) = q_{dyn}(t) S \frac{b}{2} \left(C_{l_0}(t) + C_{l_\beta}(t) \beta(t) + C_{l_p}(t) \frac{b p(t)}{2 V_a(t)} + \right. \\
\left. + C_{l_r}(t) \frac{b r(t)}{2 V_a(t)} \right) \wedge \\
M_{a_{sd}}(t) = q_{dyn}(t) S \bar{c} \left(C_{m_0}(t) + C_{m_\alpha}(t) \alpha(t) + C_{m_{\alpha^2}}(t) \alpha^2(t) + \right. \\
\left. + C_{m_q}(t) \frac{\bar{c} q(t)}{V_a(t)} + C_{m_{\beta^2}}(t) \beta^2(t) + C_{m_r}(t) \frac{b r(t)}{2 V_a(t)} \right) \wedge \\
N_{a_{sd}}(t) = q_{dyn}(t) S \frac{b}{2} \left(C_{n_0}(t) + C_{n_\beta}(t) \beta(t) + C_{n_p}(t) \frac{b p(t)}{2 V_a(t)} + \right. \\
\left. + C_{n_r}(t) \frac{b r(t)}{2 V_a(t)} + C_{n_q}(t) \frac{\bar{c} q(t)}{V(t)} + C_{n_{\beta^3}}(t) \beta^3(t) \right) \left. \right\}$$

Kinematic equation requirement

$$\mathcal{R}_{Keq} = \left\{ \left(\mathcal{X}, \mathcal{X}' \right) \middle| \forall t \left(0 \leq t < \infty \Rightarrow \right. \right. \\ \dot{\psi}(t) = \frac{q(t) \sin \phi(t) + r(t) \cos \phi(t)}{\cos \theta(t)} \wedge \\ \dot{\theta}(t) = q(t) \cos \phi(t) - r(t) \sin \phi(t) \wedge \\ \left. \left. \dot{\phi}(t) = p(t) + (q(t) \sin \phi(t) + r(t) \cos \phi(t)) \tan \theta(t) \right) \right\} \quad (\text{B.12})$$

Navigation equation requirement

$$\mathcal{R}_{Neq} = \left\{ \left((\mathcal{U}_w, \mathcal{X}), \mathcal{X} \right) \middle| \forall t \left(0 \leq t < \infty \Rightarrow \right. \right. \\ \left. \left. \dot{H}(t) = u_e(t) \sin \theta(t) + (v_e(t) \sin \phi(t) + w_e(t) \cos \phi(t)) \cos \theta(t) \right) \right\} \quad (\text{B.13})$$

Gravity force requirement

$$\mathcal{R}_{grf} = \left\{ \left(\mathcal{X}, \mathcal{C}_f \right) \middle| \forall t \exists m() \left(\right. \\ 0 \leq t < \infty \wedge m^- \leq m(t) \leq m^+ \Rightarrow \\ X_{gr}(t) = -m(t)g_0 \sin \theta(t) \wedge \\ Y_{gr}(t) = m(t)g_0 \cos \theta(t) \sin \phi(t) \wedge \\ \left. \left. Z_{gr}(t) = m(t)g_0 \cos \theta(t) \cos \phi(t) \right) \right\} \quad (\text{B.14})$$

Wind Force requirement

$$\mathcal{R}_{wf} = \left\{ \left((\mathcal{U}_w, \mathcal{X}), \mathcal{C}_f \right) \middle| \forall t \exists m() \left(\right. \\ 0 \leq t < \infty \wedge m^- \leq m(t) \leq m^+ \Rightarrow \\ X_w(t) = -m(t)(\dot{u}_w(t) + q(t)w_w(t) - r(t)v_w(t)) \wedge \\ Y_w(t) = -m(t)(\dot{v}_w(t) - p(t)w_w(t) + r(t)u_w(t)) \wedge \\ \left. \left. Z_w(t) = -m(t)(\dot{w}_w(t) + p(t)v_w(t) - q(t)u_w(t)) \right) \right\} \quad (\text{B.15})$$

Control-surface-exerted aerodynamic force requirement

$$\begin{aligned}
 \mathcal{R}_{csf} = & \quad (B.16) \\
 & \left\{ \left((\mathcal{U}_c, \mathcal{X}), \mathcal{C}_f \right) \middle| \forall t \exists C_{X_{\delta r}}() \exists C_{X_{\delta f}}() \exists C_{X_{\alpha \delta f}}() \exists C_{Y_{\delta a}}() \exists C_{Y_{\delta r}}() \exists C_{Y_{\alpha \delta r}}() \right. \\
 & \quad \exists C_{Z_{\delta e}}() \exists C_{Z_{\beta^2 \delta e}}() \exists C_{Z_{\delta f}}() \exists C_{Z_{\alpha \delta f}}() \left(\right. \\
 & \quad 0 \leq t < \infty \wedge \\
 & \quad C_{X_{\delta r}}(1 - \Delta C_{X_{\delta r}}\%) \leq C_{X_{\delta r}}(t) \leq C_{X_{\delta r}}(1 + \Delta C_{X_{\delta r}}\%) \wedge \\
 & \quad C_{X_{\delta f}}(1 - \Delta C_{X_{\delta f}}\%) \leq C_{X_{\delta f}}(t) \leq C_{X_{\delta f}}(1 + \Delta C_{X_{\delta f}}\%) \wedge \\
 & \quad C_{X_{\alpha \delta f}}(1 - \Delta C_{X_{\alpha \delta f}}\%) \leq C_{X_{\alpha \delta f}}(t) \leq C_{X_{\alpha \delta f}}(1 + \Delta C_{X_{\alpha \delta f}}\%) \wedge \\
 & \quad C_{Y_{\delta a}}(1 - \Delta C_{Y_{\delta a}}\%) \leq C_{Y_{\delta a}}(t) \leq C_{Y_{\delta a}}(1 + \Delta C_{Y_{\delta a}}\%) \wedge \\
 & \quad C_{Y_{\delta r}}(1 - \Delta C_{Y_{\delta r}}\%) \leq C_{Y_{\delta r}}(t) \leq C_{Y_{\delta r}}(1 + \Delta C_{Y_{\delta r}}\%) \wedge \\
 & \quad C_{Y_{\alpha \delta r}}(1 - \Delta C_{Y_{\alpha \delta r}}\%) \leq C_{Y_{\alpha \delta r}}(t) \leq C_{Y_{\alpha \delta r}}(1 + \Delta C_{Y_{\alpha \delta r}}\%) \wedge \\
 & \quad C_{Z_{\delta e}}(1 - \Delta C_{Z_{\delta e}}\%) \leq C_{Z_{\delta e}}(t) \leq C_{Z_{\delta e}}(1 + \Delta C_{Z_{\delta e}}\%) \wedge \\
 & \quad C_{Z_{\beta^2 \delta e}}(1 - \Delta C_{Z_{\beta^2 \delta e}}\%) \leq C_{Z_{\beta^2 \delta e}}(t) \leq C_{Z_{\beta^2 \delta e}}(1 + \Delta C_{Z_{\beta^2 \delta e}}\%) \wedge \\
 & \quad C_{Z_{\delta f}}(1 - \Delta C_{Z_{\delta f}}\%) \leq C_{Z_{\delta f}}(t) \leq C_{Z_{\delta f}}(1 + \Delta C_{Z_{\delta f}}\%) \wedge \\
 & \quad C_{Z_{\alpha \delta f}}(1 - \Delta C_{Z_{\alpha \delta f}}\%) \leq C_{Z_{\alpha \delta f}}(t) \leq C_{Z_{\alpha \delta f}}(1 + \Delta C_{Z_{\alpha \delta f}}\%) \Rightarrow \\
 & \quad X_{acd}(t) = q_{dyn}(t)S (C_{X_{\delta r}}(t)\delta_r(t) + C_{X_{\delta f}}(t)\delta_f(t) + C_{X_{\alpha \delta f}}(t)\alpha(t)\delta_f(t)) \wedge \\
 & \quad Y_{acd}(t) = q_{dyn}(t)S (C_{Y_{\delta a}}(t)\delta_a(t) + C_{Y_{\delta r}}(t)\delta_r(t) + C_{Y_{\alpha \delta r}}(t)\alpha(t)\delta_r(t)) \wedge \\
 & \quad Z_{acd}(t) = q_{dyn}(t)S (C_{Z_{\delta e}}(t)\delta_e(t) + C_{Z_{\beta^2 \delta e}}(t)\beta^2(t)\delta_e(t) + C_{Z_{\delta f}}(t)\delta_f(t)) \left. \right\}
 \end{aligned}$$

Control-surface-exerted aerodynamic moment requirement

$$\begin{aligned}
 \mathcal{R}_{csm} = & \quad (B.17) \\
 & \left\{ \left((\mathcal{U}_c, \mathcal{X}), \mathcal{C}_m \right) \middle| \forall t \exists C_{l_{\delta a}}() \exists C_{l_{\delta r}}() \exists C_{l_{\alpha \delta a}}() \exists C_{m_{\delta e}}() C_{m_{\delta f}}() \right. \\
 & \quad \exists C_{n_{\delta a}}() \exists C_{n_{\delta r}}() \left(0 \leq t < \infty \wedge \right. \\
 & \quad C_{l_{\delta a}}(1 - \Delta C_{l_{\delta a}}\%) \leq C_{l_{\delta a}}(t) \leq C_{l_{\delta a}}(1 + \Delta C_{l_{\delta a}}\%) \wedge \\
 & \quad C_{l_{\delta r}}(1 - \Delta C_{l_{\delta r}}\%) \leq C_{l_{\delta r}}(t) \leq C_{l_{\delta r}}(1 + \Delta C_{l_{\delta r}}\%) \wedge \\
 & \quad C_{l_{\alpha \delta a}}(1 - \Delta C_{l_{\alpha \delta a}}\%) \leq C_{l_{\alpha \delta a}}(t) \leq C_{l_{\alpha \delta a}}(1 + \Delta C_{l_{\alpha \delta a}}\%) \wedge \\
 & \quad C_{m_{\delta e}}(1 - \Delta C_{m_{\delta e}}\%) \leq C_{m_{\delta e}}(t) \leq C_{m_{\delta e}}(1 + \Delta C_{m_{\delta e}}\%) \wedge \\
 & \quad C_{m_{\delta f}}(1 - \Delta C_{m_{\delta f}}\%) \leq C_{m_{\delta f}}(t) \leq C_{m_{\delta f}}(1 + \Delta C_{m_{\delta f}}\%) \wedge \\
 & \quad C_{n_{\delta a}}(1 - \Delta C_{n_{\delta a}}\%) \leq C_{n_{\delta a}}(t) \leq C_{n_{\delta a}}(1 + \Delta C_{n_{\delta a}}\%) \wedge \\
 & \quad C_{n_{\delta r}}(1 - \Delta C_{n_{\delta r}}\%) \leq C_{n_{\delta r}}(t) \leq C_{n_{\delta r}}(1 + \Delta C_{n_{\delta r}}\%) \Rightarrow \\
 & \quad L_{acd}(t) = q_{dyn}(t)S \frac{b}{2} (C_{l_{\delta a}}(t)\delta_a(t) + C_{l_{\delta r}}(t)\delta_r(t) + C_{l_{\alpha \delta a}}(t)\alpha(t)\delta_a(t)) \wedge
 \end{aligned}$$

$$\begin{aligned} M_{acd}(t) &= q_{dyn}(t)S\bar{c} \left(C_{m_{\delta e}}(t)\delta_e(t) + C_{m_{\delta f}}(t)\delta_f(t) \right) \wedge \\ N_{acd}(t) &= q_{dyn}(t)S\frac{b}{2} \left(C_{n_{\delta a}}(t)\delta_a(t) + C_{n_{\delta r}}(t)\delta_r(t) \right) \Big\} \end{aligned}$$

Propulsive force requirement

$$\begin{aligned} \mathcal{R}_{pf} = & \quad (B.18) \\ \Big\{ & \left((U_c, \mathcal{X}), \mathcal{C}_f \right) \Big| \forall t \exists P() \exists dpt() \left(0 \leq t < \infty \wedge \right. \\ & P(t) = C_{e3} \left[C_{e4} + \left(C_{e5}(p_z(t) + C_{e6})(n(t) + C_{e7}) + \right. \right. \\ & \quad \left. \left. + (C_{e8} + C_{e9}n(t)) \left(1 - \frac{\rho(t)}{\rho_0} \right) \right) \right] \wedge \\ & P(t) < P^+ \wedge n^- < n(t) < n^+ \wedge p_z^- < p_z(t) < p_z^+ \wedge \\ & dpt(t) = C_{e1} + C_{e2} \left(\frac{P(t)}{0.5\rho(t)V_a^3(t)} \right) \Rightarrow \\ & X_p(t) = q_{dyn}(t)S \left[C_{X_{dpt}}dpt(t) + C_{X_{\alpha dpt^2}}\alpha(t)dpt^2(t) \right] \wedge \\ & Y_p(t) = 0 \wedge \\ & \left. Z_p(t) = q_{dyn}(t)SC_{Z_{dpt}}dpt(t) \right) \Big\} \end{aligned}$$

Propulsive moment requirement

$$\begin{aligned} \mathcal{R}_{pm} = & \quad (B.19) \\ \Big\{ & \left((U_c, \mathcal{X}), \mathcal{C}_m \right) \Big| \forall t \exists P() \exists dpt() \left(0 \leq t < \infty \wedge \right. \\ & P(t) = C_{e3} \left[C_{e4} + \left(C_{e5}(p_z(t) + C_{e6})(n(t) + C_{e7}) + \right. \right. \\ & \quad \left. \left. + (C_{e8} + C_{e9}n(t)) \left(1 - \frac{\rho(t)}{\rho_0} \right) \right) \right] \wedge \\ & P(t) < P^+ \wedge n^- < n(t) < n^+ \wedge p_z^- < p_z(t) < p_z^+ \wedge \\ & dpt(t) = C_{e1} + C_{e2} \left(\frac{P(t)}{0.5\rho(t)V_a^3(t)} \right) \Rightarrow \\ & L_p(t) = q_{dyn}(t)S\frac{b}{2}C_{l_{\alpha^2 dpt}}\alpha^2(t)dpt(t) \wedge \\ & M_p(t) = q_{dyn}(t)S\bar{c}C_{m_{dpt}}dpt(t) \wedge \\ & \left. N_p(t) = q_{dyn}(t)S\frac{b}{2}C_{n_{dpt^3}}dpt^3(t) \right) \Big\} \end{aligned}$$

Kinematic acceleration at crew station requirement

$$\mathcal{R}_{ka} = \left\{ \left((\mathcal{U}_w, \mathcal{X}), \mathcal{X}' \right) \mid \forall t \left(0 \leq t < \infty \Rightarrow \right. \right. \\
\begin{aligned}
& A_x(t) = \dot{u}_e(t) + q(t)w_e(t) - r(t)v_e(t) + \dot{q}(t)r_z - \dot{r}(t)r_y + \\
& \quad + p(t)q(t)r_y + p(t)r(t)r_z - q^2(t)r_x - r^2(t)r_x \wedge \\
& A_y(t) = \dot{v}_e(t) + r(t)u_e(t) - p(t)w_e(t) + \dot{r}(t)r_x - \dot{p}(t)r_z + \\
& \quad + p(t)q(t)r_x + q(t)r(t)r_z - p^2(t)r_y - r^2(t)r_y \wedge \\
& A_z(t) = \dot{w}_e(t) + p(t)v_e(t) - q(t)u_e(t) + \dot{p}(t)r_y - \dot{q}(t)r_x + \\
& \quad \left. \left. + p(t)r(t)r_x + q(t)r(t)r_y - p^2(t)r_z - q^2(t)r_z \right) \right\} \quad (B.20)
\end{aligned}$$

Air data requirement

$$\mathcal{R}_{ad} = \left\{ \left(\mathcal{X}, \mathcal{X}' \right) \mid \forall t \left(0 \leq t < \infty \Rightarrow \right. \right. \\
\begin{aligned}
& T(t) = T_0 + \lambda H(t) \wedge \\
& p_s(t) = p_0 \left(\frac{T_0}{T(t)} \right)^{\frac{\gamma_0}{\lambda R}} \wedge \\
& \rho(t) = \frac{p_s(t)}{RT(t)} \wedge \\
& \left. q_{dyn}(t) = 0.5\rho(t)V_a^2(t) \right\} \quad (B.21)
\end{aligned}$$

B.2.2 DHC-2 Flight Control System Hardware

Rudder actuator

$$\mathcal{R}_{rud} = \left\{ \left((\mathcal{X}, \tilde{\mathcal{U}}_c), \mathcal{U}_c \right) \mid \forall t \exists \mathbf{x}_r() \left(0 \leq t < \infty \wedge \right. \right. \\
\begin{aligned}
& \mathbf{x}_r(0) = -\mathbf{A}_r^{-1}\mathbf{B}_r\tilde{\mathbf{u}}_r(0) \wedge \\
& \dot{\mathbf{x}}_r(t) = \mathbf{A}_r\mathbf{x}_r(t) + \mathbf{B}_r\tilde{\mathbf{u}}_r(t) \Rightarrow \\
& \left. \delta_r(t) = \mathbf{C}_r\mathbf{x}_r(t) + \mathbf{D}_r\tilde{\mathbf{u}}_r(t) \right\} \quad (B.22)
\end{aligned}$$

Aileron actuator

$$\mathcal{R}_{ail} = \left\{ \left((\mathcal{X}, \tilde{\mathcal{U}}_c), \mathcal{U}_c \right) \middle| \forall t \exists \mathbf{x}_a() \left(0 \leq t < \infty \wedge \right. \right. \\ \left. \left. \begin{aligned} \mathbf{x}_a(0) &= -\mathbf{A}_a^{-1} \mathbf{B}_a \tilde{\mathbf{u}}_a(0) \wedge \\ \dot{\mathbf{x}}_a(t) &= \mathbf{A}_a \mathbf{x}_a(t) + \mathbf{B}_a \tilde{\mathbf{u}}_a(t) \Rightarrow \\ \delta_a(t) &= 2\mathbf{C}_a \mathbf{x}_a(t) + \mathbf{D}_a \tilde{\mathbf{u}}_a(t) \end{aligned} \right) \right\} \quad (\text{B.23})$$

Elevator actuator

$$\mathcal{R}_{elv} = \left\{ \left((\mathcal{X}, \tilde{\mathcal{U}}_c), \mathcal{U}_c \right) \middle| \forall t \exists \mathbf{x}_e() \left(0 \leq t < \infty \wedge \right. \right. \\ \left. \left. \begin{aligned} \mathbf{x}_e(0) &= -\mathbf{A}_e^{-1} \mathbf{B}_e \tilde{\mathbf{u}}_e(0) \wedge \\ \dot{\mathbf{x}}_e(t) &= \mathbf{A}_e \mathbf{x}_e(t) + \mathbf{B}_e \tilde{\mathbf{u}}_e(t) \Rightarrow \\ \delta_e(t) &= \mathbf{C}_e \mathbf{x}_e(t) + \mathbf{D}_e \tilde{\mathbf{u}}_e(t) \end{aligned} \right) \right\} \quad (\text{B.24})$$

Roll rate gyro requirement

$$\mathcal{R}_p = \left\{ \left(\mathcal{X}, \tilde{\mathcal{X}} \right) \middle| \forall t \exists S_p \exists bias_p \exists \nu_p() \left(0 \leq t < \infty \wedge \right. \right. \\ \left. \left. \begin{aligned} S_g^- &\leq S_p \leq S_g^+ \wedge \\ BIAS_g^- &\leq bias_p \leq BIAS_g^+ \wedge \\ whiteNoise(\nu_p(), Npsd_g) &\Rightarrow \\ \tilde{p}(t) &\simeq \left[S_p \left[p(t) \right]_{IR_g^-}^{IR_g^+} + bias_p + \nu_p(t) \right]_{OR_g^-}^{OR_g^+} \end{aligned} \right) \right\} \quad (\text{B.25})$$

Pitch rate gyro requirement

$$\mathcal{R}_q = \left\{ \left(\mathcal{X}, \tilde{\mathcal{X}} \right) \middle| \forall t \exists S_q \exists bias_q \exists \nu_q() \left(0 \leq t < \infty \wedge \right. \right. \\ \left. \left. \begin{aligned} S_g^- &\leq S_q \leq S_g^+ \wedge \\ BIAS_g^- &\leq bias_q \leq BIAS_g^+ \wedge \end{aligned} \right) \right\} \quad (\text{B.26})$$

$$whiteNoise(\nu_q(), Npsd_g) \Rightarrow$$

$$\tilde{q}(t) \simeq \left[S_q [q(t)]_{IR_g^-}^{IR_g^+} + bias_q + \nu_q(t) \right]_{OR_g^-}^{OR_g^+} \Bigg\}$$

Yaw rate gyro requirement

$$\mathcal{R}_r = \tag{B.27}$$

$$\left\{ \left(\mathcal{X}, \tilde{\mathcal{X}} \right) \middle| \forall t \exists S_r \exists bias_r \exists \nu_r() \left(0 \leq t < \infty \wedge \right. \right.$$

$$S_g^- \leq S_r \leq S_g^+ \wedge$$

$$BIAS_g^- \leq bias_r \leq BIAS_g^+ \wedge$$

$$whiteNoise(\nu_r(), Npsd_g) \Rightarrow$$

$$\left. \tilde{r}(t) \simeq \left[S_r [r(t)]_{IR_g^-}^{IR_g^+} + bias_r + \nu_r(t) \right]_{OR_g^-}^{OR_g^+} \right\}$$

Angle of attack sensor requirement

$$\mathcal{R}_\alpha = \tag{B.28}$$

$$\left\{ \left(\mathcal{X}, \tilde{\mathcal{X}} \right) \middle| \forall t \exists S_\alpha \exists bias_\alpha \exists \nu_\alpha() \left(0 \leq t < \infty \wedge \right. \right.$$

$$S_\alpha^- \leq S_\alpha \leq S_\alpha^+ \wedge$$

$$BIAS_\alpha^- \leq bias_\alpha \leq BIAS_\alpha^+ \wedge$$

$$whiteNoise(\nu_\alpha(), Npsd_\alpha) \Rightarrow$$

$$\left. \tilde{\alpha}(t) \simeq \left[S_\alpha [\alpha(t)]_{IR_\alpha^-}^{IR_\alpha^+} + bias_\alpha + \nu_\alpha(t) \right]_{OR_\alpha^-}^{OR_\alpha^+} \right\}$$

A_x accelerometer requirement

$$\mathcal{R}_{A_x} = \tag{B.29}$$

$$\left\{ \left(\mathcal{X}, \tilde{\mathcal{X}} \right) \middle| \forall t \exists S_{A_x} \exists bias_{A_x} \exists \nu_{A_x}() \left(0 \leq t < \infty \wedge \right. \right.$$

$$S_A^- \leq S_{A_x} \leq S_A^+ \wedge$$

$$BIAS_A^- \leq bias_{A_x} \leq BIAS_A^+ \wedge$$

$$whiteNoise(\nu_{A_x}(), Npsd_A) \Rightarrow$$

$$\left. \tilde{A}_x(t) \simeq \left[S_{A_x} [A_x(t) + \sin \theta(t)]_{IR_A^-}^{IR_A^+} + bias_{A_x} + \nu_{A_x}(t) \right]_{OR_A^-}^{OR_A^+} \right\}$$

A_y accelerometer requirement

$$\begin{aligned} \mathcal{R}_{A_y} = & \quad (B.30) \\ & \left\{ \left(\mathcal{X}, \tilde{\mathcal{X}} \right) \middle| \forall t \exists S_{A_y} \exists bias_{A_y} \exists \nu_{A_y}() \left(0 \leq t < \infty \wedge \right. \right. \\ & \quad S_A^- \leq S_{A_y} \leq S_A^+ \wedge \\ & \quad BIAS_A^- \leq bias_{A_y} \leq BIAS_A^+ \wedge \\ & \quad whiteNoise(\nu_{A_y}(), Npsd_A) \Rightarrow \\ & \quad \left. \tilde{A}_y(t) \simeq \left[S_{A_y} \left[A_y(t) - \cos \theta(t) \sin \phi(t) \right]_{IR_A^-}^{IR_A^+} + bias_{A_y} + \nu_{A_y}(t) \right]_{OR_A^-}^{OR_A^+} \right\} \end{aligned}$$

A_z accelerometer requirement

$$\begin{aligned} \mathcal{R}_{A_z} = & \quad (B.31) \\ & \left\{ \left(\mathcal{X}, \tilde{\mathcal{X}} \right) \middle| \forall t \exists S_{A_z} \exists bias_{A_z} \exists \nu_{A_z}() \left(0 \leq t < \infty \wedge \right. \right. \\ & \quad S_A^- \leq S_{A_z} \leq S_A^+ \wedge \\ & \quad BIAS_A^- \leq bias_{A_z} \leq BIAS_A^+ \wedge \\ & \quad whiteNoise(\nu_{A_z}(), Npsd_A) \Rightarrow \\ & \quad \left. \tilde{A}_z(t) \simeq \left[S_{A_z} \left[A_z(t) + \cos \theta(t) \sin \phi(t) \right]_{IR_A^-}^{IR_A^+} + bias_{A_z} + \nu_{A_z}(t) \right]_{OR_A^-}^{OR_A^+} \right\} \end{aligned}$$

Dynamic pressure sensor requirement

$$\begin{aligned} \mathcal{R}_{q_{dyn}} = & \quad (B.32) \\ & \left\{ \left(\mathcal{X}, \tilde{\mathcal{X}} \right) \middle| \forall t \exists S_{q_{dyn}} \exists bias_{q_{dyn}} \exists \nu_{q_{dyn}}() \left(0 \leq t < \infty \wedge \right. \right. \\ & \quad S_{q_{dyn}}^- \leq S_{q_{dyn}} \leq S_{q_{dyn}}^+ \wedge \\ & \quad BIAS_{q_{dyn}}^- \leq bias_{q_{dyn}} \leq BIAS_{q_{dyn}}^+ \wedge \\ & \quad whiteNoise(\nu_{q_{dyn}}(), Npsd_{q_{dyn}}) \Rightarrow \\ & \quad \left. \tilde{q}_{dyn}(t) \simeq \left[S_{q_{dyn}} \left[q_{dyn}(t) \right]_{IR_{q_{dyn}}^-}^{IR_{q_{dyn}}^+} + bias_{q_{dyn}} + \nu_{q_{dyn}}(t) \right]_{OR_{q_{dyn}}^-}^{OR_{q_{dyn}}^+} \right\} \end{aligned}$$

Static pressure sensor requirement

$$\begin{aligned} \mathcal{R}_{p_s} = & \quad (B.33) \\ & \left\{ \left(\mathcal{X}, \tilde{\mathcal{X}} \right) \middle| \forall t \exists S_{p_s} \exists bias_{p_s} \exists \nu_{p_s}() \left(0 \leq t < \infty \wedge \right. \right. \end{aligned}$$

$$\begin{aligned}
& S_{p_s}^- \leq S_{p_s} \leq S_{p_s}^+ \wedge \\
& BIAS_{p_s}^- \leq bias_{p_s} \leq BIAS_{p_s}^+ \wedge \\
& whiteNoise(\nu_{p_s}(), Npsd_{p_s}) \Rightarrow \\
& \tilde{p}_s(t) \simeq \left[S_{p_s} [p_s(t)]_{IR_{p_s}^-}^{IR_{p_s}^+} + bias_{p_s} + \nu_{p_s}(t) \right]_{OR_{p_s}^-}^{OR_{p_s}^+} \Bigg\}
\end{aligned}$$

Temperature sensor requirement

$$\begin{aligned}
\mathcal{R}_T = & \tag{B.34} \\
& \left\{ \left(\mathcal{X}, \tilde{\mathcal{X}} \right) \middle| \forall t \exists S_T \exists bias_T \exists \nu_T() \left(0 \leq t < \infty \wedge \right. \right. \\
& S_T^- \leq S_T \leq S_T^+ \wedge \\
& BIAS_T^- \leq bias_T \leq BIAS_T^+ \wedge \\
& whiteNoise(\nu_T(), Npsd_T) \Rightarrow \\
& \left. \tilde{T}(t) \simeq \left[S_T [T(t)]_{IR_T^-}^{IR_T^+} + bias_T + \nu_T(t) \right]_{OR_T^-}^{OR_T^+} \right\}
\end{aligned}$$

Pitch attitude sensor requirement

$$\begin{aligned}
\mathcal{R}_\theta = & \tag{B.35} \\
& \left\{ \left(\mathcal{X}, \tilde{\mathcal{X}} \right) \middle| \forall t \exists S_\theta \exists bias_\theta \exists \nu_\theta() \left(0 \leq t < \infty \wedge \right. \right. \\
& S_{\phi/\theta}^- \leq S_\theta \leq S_{\phi/\theta}^+ \wedge \\
& BIAS_{\phi/\theta}^- \leq bias_\theta \leq BIAS_{\phi/\theta}^+ \wedge \\
& whiteNoise(\nu_\theta(), Npsd_{\phi/\theta}) \Rightarrow \\
& \left. \tilde{\theta}(t) \simeq \left[S_\theta [\theta(t)]_{IR_{\phi/\theta}^-}^{IR_{\phi/\theta}^+} + bias_\theta + \nu_\theta(t) \right]_{OR_{\phi/\theta}^-}^{OR_{\phi/\theta}^+} \right\}
\end{aligned}$$

Roll attitude sensor requirement

$$\begin{aligned}
\mathcal{R}_\phi = & \tag{B.36} \\
& \left\{ \left(\mathcal{X}, \tilde{\mathcal{X}} \right) \middle| \forall t \exists S_\phi \exists bias_\phi \exists \nu_\phi() \left(0 \leq t < \infty \wedge \right. \right. \\
& S_{\phi/\theta}^- \leq S_\phi \leq S_{\phi/\theta}^+ \wedge \\
& BIAS_{\phi/\theta}^- \leq bias_\phi \leq BIAS_{\phi/\theta}^+ \wedge \\
& whiteNoise(\nu_\phi(), Npsd_{\phi/\theta}) \Rightarrow \\
& \left. \tilde{\phi}(t) \simeq \left[S_\phi [\phi(t)]_{IR_{\phi/\theta}^-}^{IR_{\phi/\theta}^+} + bias_\phi + \nu_\phi(t) \right]_{OR_{\phi/\theta}^-}^{OR_{\phi/\theta}^+} \right\}
\end{aligned}$$

Heading sensor requirement

$$\mathcal{R}_\psi = \left\{ \left(\mathcal{X}, \tilde{\mathcal{X}} \right) \middle| \forall t \exists S_\psi \exists bias_\psi \exists \nu_\psi() \left(0 \leq t < \infty \wedge \right. \right. \\ \left. S_\psi^- \leq S_\psi \leq S_\psi^+ \wedge \right. \\ \left. BIAS_\psi^- \leq bias_\psi \leq BIAS_\psi^+ \wedge \right. \\ \left. whiteNoise(\nu_\psi(), Npsd_\psi) \Rightarrow \right. \\ \left. \tilde{\psi}(t) \simeq \left[S_\psi \left[\psi(t) \right]_{IR_\psi^-}^{IR_\psi^+} + bias_\psi + \nu_\psi(t) \right]_{OR_\psi^-}^{OR_\psi^+} \right\} \quad (\text{B.37})$$

DAC card requirement

$$\mathcal{R}_{DAC} = \left\{ \left(\bar{\mathcal{U}}_c, \tilde{\mathcal{U}}_c \right) \middle| \forall k \forall t \left(\right. \\ \left. 0 \leq k < \infty \wedge kT_s \leq t < (k+1)T_s \Rightarrow \right. \\ \left. \tilde{\delta}_e(t) = (\bar{\delta}_e(k) - OFF_{DAC})S_{DAC}^{-1} \wedge \right. \\ \left. \tilde{\delta}_a(t) = (\bar{\delta}_a(k) - OFF_{DAC})S_{DAC}^{-1} \wedge \right. \\ \left. \tilde{\delta}_r(t) = (\bar{\delta}_r(k) - OFF_{DAC})S_{DAC}^{-1} \right\} \quad (\text{B.38})$$

ADC card requirement

$$\mathcal{R}_{ADC} = \left\{ \left((\tilde{\mathcal{U}}_p, \tilde{\mathcal{X}}), (\bar{\mathcal{U}}_p, \bar{\mathcal{X}}) \right) \middle| \forall k \left(0 \leq k < \infty \Rightarrow \right. \right. \\ \left. \bar{p}(k) = \left[S_{ADC} \left[\tilde{p}(kT_s) \right]_{ADC_{V^-}}^{ADC_{V^+}} + OFF_{ADC} \right] \wedge \right. \\ \left. \bar{q}(k) = \left[S_{ADC} \left[\tilde{q}(kT_s) \right]_{ADC_{V^-}}^{ADC_{V^+}} + OFF_{ADC} \right] \wedge \right. \\ \left. \bar{r}(k) = \left[S_{ADC} \left[\tilde{r}(kT_s) \right]_{ADC_{V^-}}^{ADC_{V^+}} + OFF_{ADC} \right] \wedge \right. \\ \left. \bar{\alpha}(k) = \left[S_{ADC} \left[\tilde{\alpha}(kT_s) \right]_{ADC_{V^-}}^{ADC_{V^+}} + OFF_{ADC} \right] \wedge \right. \\ \left. \bar{A}_x(k) = \left[S_{ADC} \left[\tilde{A}_x(kT_s) \right]_{ADC_{V^-}}^{ADC_{V^+}} + OFF_{ADC} \right] \wedge \right. \\ \left. \bar{A}_y(k) = \left[S_{ADC} \left[\tilde{A}_y(kT_s) \right]_{ADC_{V^-}}^{ADC_{V^+}} + OFF_{ADC} \right] \wedge \right. \\ \left. \bar{A}_z(k) = \left[S_{ADC} \left[\tilde{A}_z(kT_s) \right]_{ADC_{V^-}}^{ADC_{V^+}} + OFF_{ADC} \right] \wedge \right. \\ \left. \bar{\psi}(k) = \left[S_{ADC} \left[\tilde{\psi}(kT_s) \right]_{ADC_{V^-}}^{ADC_{V^+}} + OFF_{ADC} \right] \wedge \right. \\ \left. \bar{\theta}(k) = \left[S_{ADC} \left[\tilde{\theta}(kT_s) \right]_{ADC_{V^-}}^{ADC_{V^+}} + OFF_{ADC} \right] \wedge \right. \\ \left. \right\} \quad (\text{B.39})$$

$$\begin{aligned}
\bar{\phi}(k) &= \left[S_{ADC} [\tilde{\phi}(k T_s)]_{ADC_{V^-}}^{ADC_{V^+}} + OFF_{ADC} \right] \wedge \\
\bar{q}_{dyn}(k) &= \left[S_{ADC} [\tilde{q}_{dyn}(k T_s)]_{ADC_{V^-}}^{ADC_{V^+}} + OFF_{ADC} \right] \wedge \\
\bar{p}_s(k) &= \left[S_{ADC} [\tilde{p}_s(k T_s)]_{ADC_{V^-}}^{ADC_{V^+}} + OFF_{ADC} \right] \wedge \\
\bar{T}(k) &= \left[S_{ADC} [\tilde{T}(k T_s)]_{ADC_{V^-}}^{ADC_{V^+}} + OFF_{ADC} \right] \wedge \\
\bar{\phi}_r(k) &= \left[S_{ADC} [\tilde{\phi}_r(k T_s)]_{ADC_{V^-}}^{ADC_{V^+}} + OFF_{ADC} \right] \wedge \\
\bar{\psi}_r(k) &= \left[S_{ADC} [\tilde{\psi}_r(k T_s)]_{ADC_{V^-}}^{ADC_{V^+}} + OFF_{ADC} \right] \wedge \\
\bar{\theta}_r(k) &= \left[S_{ADC} [\tilde{\theta}_r(k T_s)]_{ADC_{V^-}}^{ADC_{V^+}} + OFF_{ADC} \right] \wedge \\
\bar{S}W_{PAH}(k) &= \left[S_{ADC} [\widetilde{S}W_{PAH}(k T_s)]_{ADC_{V^-}}^{ADC_{V^+}} + OFF_{ADC} \right] \wedge \\
\bar{S}W_{ALH}(k) &= \left[S_{ADC} [\widetilde{S}W_{ALH}(k T_s)]_{ADC_{V^-}}^{ADC_{V^+}} + OFF_{ADC} \right] \wedge \\
\bar{S}W_{RAH}(k) &= \left[S_{ADC} [\widetilde{S}W_{RAH}(k T_s)]_{ADC_{V^-}}^{ADC_{V^+}} + OFF_{ADC} \right] \wedge \\
\bar{S}W_{HH}(k) &= \left[S_{ADC} [\widetilde{S}W_{HH}(k T_s)]_{ADC_{V^-}}^{ADC_{V^+}} + OFF_{ADC} \right] \wedge \\
\bar{S}W_{HS}(k) &= \left[S_{ADC} [\widetilde{S}W_{HS}(k T_s)]_{ADC_{V^-}}^{ADC_{V^+}} + OFF_{ADC} \right] \wedge
\end{aligned}$$

Control panel requirement

$$\begin{aligned}
\mathcal{R}_{CP} = & \quad (B.40) \\
& \left\{ \left(\mathcal{U}_p, \tilde{\mathcal{U}}_p \right) \mid \forall t \left(0 \leq t < \infty \Rightarrow \right. \right. \\
& \quad (\widetilde{S}W_{PAH}(t) > V_{th} \Leftrightarrow SW_{PAH}(t) = ON) \wedge \\
& \quad (\widetilde{S}W_{ALH}(t) > V_{th} \Leftrightarrow SW_{ALH}(t) = ON) \wedge \\
& \quad (\widetilde{S}W_{RAH}(t) > V_{th} \Leftrightarrow SW_{RAH}(t) = ON) \wedge \\
& \quad (\widetilde{S}W_{HH}(t) > V_{th} \Leftrightarrow SW_{HH}(t) = ON) \wedge \\
& \quad (\widetilde{S}W_{HS}(t) > V_{th} \Leftrightarrow SW_{HS}(t) = ON) \wedge \\
& \quad \tilde{\theta}_r(t) = S_{\theta_r} \theta_r(t) + BIAS_{\theta_r} \wedge \\
& \quad \tilde{\phi}_r(t) = S_{\phi_r} \phi_r(t) + BIAS_{\phi_r} \wedge \\
& \quad \left. \left. \tilde{\psi}_r(t) = S_{\psi_r} \psi_r(t) + BIAS_{\psi_r} \right) \right\}
\end{aligned}$$

B.2.3 DHC-2 Flight Control System Software

Input interface requirement

$$\begin{aligned}
 \mathcal{R}_{in} = & \quad (B.41) \\
 & \left\{ \left((\bar{\mathcal{U}}_p, \bar{\mathcal{X}}), (\hat{\mathcal{U}}_p, \hat{\mathcal{X}}) \right) \mid \forall k \left(0 \leq k < \infty \Rightarrow \right. \right. \\
 & \quad \hat{p}(k) = S_p^{-1} \left((\bar{p}(k) - OFF_{ADC})S_{ADC}^{-1} - BIAS_p \right) \wedge \\
 & \quad \hat{q}(k) = S_q^{-1} \left((\bar{q}(k) - OFF_{ADC})S_{ADC}^{-1} - BIAS_q \right) \wedge \\
 & \quad \hat{r}(k) = S_r^{-1} \left((\bar{r}(k) - OFF_{ADC})S_{ADC}^{-1} - BIAS_r \right) \wedge \\
 & \quad \hat{\alpha}(k) = S_\alpha^{-1} \left((\bar{\alpha}(k) - OFF_{ADC})S_{ADC}^{-1} - BIAS_\alpha \right) \wedge \\
 & \quad \hat{A}_x(k) = S_{A_x}^{-1} \left((\bar{A}_x(k) - OFF_{ADC})S_{ADC}^{-1} - BIAS_{A_x} \right) \wedge \\
 & \quad \hat{A}_y(k) = S_{A_y}^{-1} \left((\bar{A}_y(k) - OFF_{ADC})S_{ADC}^{-1} - BIAS_{A_y} \right) \wedge \\
 & \quad \hat{A}_z(k) = S_{A_z}^{-1} \left((\bar{A}_z(k) - OFF_{ADC})S_{ADC}^{-1} - BIAS_{A_z} \right) \wedge \\
 & \quad \hat{\psi}(k) = S_\psi^{-1} \left((\bar{\psi}(k) - OFF_{ADC})S_{ADC}^{-1} - BIAS_\psi \right) \wedge \\
 & \quad \hat{\theta}(k) = S_\theta^{-1} \left((\bar{\theta}(k) - OFF_{ADC})S_{ADC}^{-1} - BIAS_\theta \right) \wedge \\
 & \quad \hat{\phi}(k) = S_\phi^{-1} \left((\bar{\phi}(k) - OFF_{ADC})S_{ADC}^{-1} - BIAS_\phi \right) \wedge \\
 & \quad \hat{q}_{dyn}(k) = S_{q_{dyn}}^{-1} \left((\bar{q}_{dyn}(k) - OFF_{ADC})S_{ADC}^{-1} - BIAS_{q_{dyn}} \right) \wedge \\
 & \quad \hat{p}_s(k) = S_{p_s}^{-1} \left((\bar{p}_s(k) - OFF_{ADC})S_{ADC}^{-1} - BIAS_{p_s} \right) \wedge \\
 & \quad \hat{T}(k) = S_T^{-1} \left((\bar{T}(k) - OFF_{ADC})S_{ADC}^{-1} - BIAS_T \right) \wedge \\
 & \quad \hat{\phi}_r(k) = S_{\phi_r}^{-1} \left((\bar{\phi}_r(k) - OFF_{ADC})S_{ADC}^{-1} - BIAS_{\phi_r} \right) \wedge \\
 & \quad \hat{\psi}_r(k) = S_{\psi_r}^{-1} \left((\bar{\psi}_r(k) - OFF_{ADC})S_{ADC}^{-1} - BIAS_{\psi_r} \right) \wedge \\
 & \quad \hat{\theta}_r(k) = S_{\theta_r}^{-1} \left((\bar{\theta}_r(k) - OFF_{ADC})S_{ADC}^{-1} - BIAS_{\theta_r} \right) \wedge \\
 & \quad (\widehat{SW}_{PAH}(k) = ON \Leftrightarrow (\bar{SW}_{PAH}(k) - OFF_{ADC})S_{ADC}^{-1} > V_{th}) \wedge \\
 & \quad (\widehat{SW}_{ALH}(k) = ON \Leftrightarrow (\bar{SW}_{ALH}(k) - OFF_{ADC})S_{ADC}^{-1} > V_{th}) \wedge \\
 & \quad (\widehat{SW}_{RAH}(k) = ON \Leftrightarrow (\bar{SW}_{RAH}(k) - OFF_{ADC})S_{ADC}^{-1} > V_{th}) \wedge \\
 & \quad (\widehat{SW}_{HH}(k) = ON \Leftrightarrow (\bar{SW}_{HH}(k) - OFF_{ADC})S_{ADC}^{-1} > V_{th}) \wedge \\
 & \quad (\widehat{SW}_{HS}(k) = ON \Leftrightarrow (\bar{SW}_{HS}(k) - OFF_{ADC})S_{ADC}^{-1} > V_{th}) \wedge \\
 & \quad \hat{V}_a(k) = \sqrt{2\hat{q}_{dyn}(k) \frac{R\hat{T}(k)}{\hat{p}_s(k)}} \wedge \\
 & \quad \left. \hat{H}(k) = psToHtable(\hat{p}_s(k)) \right\}
 \end{aligned}$$

Output interface requirement

$$\mathcal{R}_{out} = \left\{ \left(\hat{\mathcal{U}}_c, \bar{\mathcal{U}}_c \right) \mid \forall k \left(0 \leq k < \infty \Rightarrow \right. \right. \\ \left. \bar{\delta}_e(k) = \left[S_{DAC} \left[\hat{\delta}_e(k) \right]_{DAC_{V-}}^{DAC_{V+}} + OFF_{DAC} \right] \wedge \right. \\ \left. \bar{\delta}_a(k) = \left[S_{DAC} \left[\hat{\delta}_a(k) \right]_{DAC_{V-}}^{DAC_{V+}} + OFF_{DAC} \right] \wedge \right. \\ \left. \left. \bar{\delta}_r(k) = \left[S_{DAC} \left[\hat{\delta}_r(k) \right]_{DAC_{V-}}^{DAC_{V+}} + OFF_{DAC} \right] \right\} \quad (\text{B.42})$$

Pitch Attitude Hold FCL requirement

$$\mathcal{R}_{\widehat{PAH}} = \left\{ \left((\hat{\mathcal{U}}_p, \hat{\mathcal{X}}), \hat{\mathcal{U}}_c \right) \mid \forall k \left(1 \leq k < \infty \wedge \widehat{SW}_{PAH}(k) = ON \Rightarrow \right. \right. \\ \left. \widehat{\Delta\delta}_e(k) = \left[\left(\left[\widehat{\Delta\theta}_r(k) \right]_{\Delta\theta_r^-}^{\Delta\theta_r^+} - \widehat{\Delta\theta}(k) \right) K_{\theta}(\hat{V}_a) \right]_{\Delta\theta_V^-}^{\Delta\theta_V^+} - \right. \\ \left. - \left(\hat{q}(k) - \hat{r}(k) \tan \hat{\phi}(k) - K_{tc}(\hat{V}_a)(\sec(\hat{\phi}(k) + \Delta\phi_{PAH}) - 1) \right) K_q(\hat{V}_a) + \right. \\ \left. + \left[\left(\left[\widehat{\Delta\theta}_r(k) \right]_{\Delta\theta_r^-}^{\Delta\theta_r^+} - \widehat{\Delta\theta}(k) \right) K_{\theta}(\hat{V}_a) \right]_{\Delta\theta_V^-}^{\Delta\theta_V^+} K_{s-i} \frac{T_s}{\mathbf{z} - 1} \right]_{l_{s1}^-}^{l_{s1}^+} \right\} \quad (\text{B.43})$$

Altitude Hold FCL requirement

$$\mathcal{R}_{\widehat{ALH}} = \left\{ \left((\hat{\mathcal{U}}_p, \hat{\mathcal{X}}), \hat{\mathcal{U}}_c \right) \mid \forall k_1 \forall k_2 \left(1 \leq k_1 < k_2 < \infty \wedge \right. \right. \\ \left. engaged(\widehat{SW}_{ALH}(), k_1, k_2) \Rightarrow \right. \\ \left. \forall k \left(1 \leq k < \infty \wedge \widehat{SW}_{ALH}(k) = ON \Rightarrow \right. \right. \\ \left. \widehat{\Delta\delta}_e(k) = \left[\left(\left[\widehat{\Delta H}(k_1) - \widehat{\Delta H}(k) \right] K_H(\hat{V}_a) \bar{K}_{\theta}^{-1}(\hat{V}_a) \right]_{\Delta\theta_r^-}^{\Delta\theta_r^+} + \right. \right. \\ \left. - \left(\left[\frac{K_d(\hat{V}_a)T_s}{\mathbf{z} - 1 + K_d(\hat{V}_a)T_s} \widehat{\Delta\theta}(k) \right]_{l_{s1}^-}^{l_{s1}^+} + \widehat{\Delta\theta}(k) \right) \bar{K}_{\theta}(\hat{V}_a) \right]_{\Delta\theta_V^-}^{\Delta\theta_V^+} \\ \left. - \left(\hat{q}(k) - \hat{r}(k) \tan \hat{\phi}(k) - \bar{K}_{tc}(\hat{V}_a)(\sec(\hat{\phi}(k) + \Delta\phi_{AL}) - 1) \right) K_q(\hat{V}_a) \right. \\ \left. + \left[\left(\left[\widehat{\Delta H}(k_1) - \widehat{\Delta H}(k) \right] K_H(\hat{V}_a) \bar{K}_{\theta}^{-1}(\hat{V}_a) \right]_{\Delta\theta_r^-}^{\Delta\theta_r^+} \right. \right. \\ \left. - \left(\left[\frac{K_d(\hat{V}_a)T_s}{\mathbf{z} - 1 + K_d(\hat{V}_a)T_s} \widehat{\Delta\theta}(k) \right]_{l_{s2}^-}^{l_{s2}^+} + \widehat{\Delta\theta}(k) \right) \bar{K}_{\theta}(\hat{V}_a) \right]_{\Delta\theta_V^-}^{\Delta\theta_V^+} \right. \\ \left. \left. \right\} \quad (\text{B.44})$$

$$\cdot K_{s-i} \frac{T_s}{\mathbf{z} - 1} \Big]_{l_{s1}^-}^{l_{s1}^+} \Big\}$$

Roll Attitude Hold FCL requirement

$$\begin{aligned} \mathcal{R}_{\widehat{RAH}} = & \quad (B.45) \\ & \left\{ \left((\hat{\mathcal{U}}_p, \hat{\mathcal{X}}), \hat{\mathcal{U}}_c \right) \mid \forall k \left(1 \leq k < \infty \wedge \widehat{SW}_{RAH}(k) = ON \Rightarrow \right. \right. \\ & \quad \widehat{\Delta\delta}_a(k) = \left[\left(\left[\widehat{\Delta\phi}_r(k) \right]_{\Delta\phi_r^-}^{\Delta\phi_r^+} - \widehat{\Delta\phi}(k) \right) K_\phi(\hat{V}_a) \right]_{\Delta\phi_V^-}^{\Delta\phi_V^+} + dar \cdot \hat{r}(k) + \\ & \quad + \left[\left(\left[\widehat{\Delta\phi}_r(k) \right]_{\Delta\phi_r^-}^{\Delta\phi_r^+} - \widehat{\Delta\phi}(k) \right) K_\phi(\hat{V}_a) \right]_{\Delta\phi_V^-}^{\Delta\phi_V^+} K_{a-i} \frac{T_s}{\mathbf{z} - 1} \Big]_{l_a^-}^{l_a^+} \wedge \\ & \quad \left. \widehat{\Delta\delta}_{\hat{r}}(k) = K_r \frac{g_0}{\hat{V}_a(k)} \sin \hat{\phi}(k) + (drr(\hat{V}_a) - K_r) \hat{r}(k) \right\} \end{aligned}$$

Heading Select FCL requirement

$$\begin{aligned} \mathcal{R}_{\widehat{HS}} = & \quad (B.46) \\ & \left\{ \left((\hat{\mathcal{U}}_p, \hat{\mathcal{X}}), \hat{\mathcal{U}}_c \right) \mid \forall k \left(1 \leq k < \infty \wedge \widehat{SW}_{HS}(k) = ON \Rightarrow \right. \right. \\ & \quad \widehat{\Delta\delta}_a(k) = \left[\left(\left[\widehat{\Delta\psi}_r(k) - \widehat{\Delta\psi}(k) \right] K_\psi(\hat{V}_a) \right]_{\Delta\phi_r^-}^{\Delta\phi_r^+} - \right. \\ & \quad \left. - \widehat{\Delta\phi}(k) \right) K_\phi(\hat{V}_a) \right]_{\Delta\phi_V^-}^{\Delta\phi_V^+} + dar \cdot \hat{r}(k) + \\ & \quad + \left[\left(\left[\left(\widehat{\Delta\psi}_r(k) - \widehat{\Delta\psi}(k) \right) K_\psi(\hat{V}_a) \right]_{\Delta\phi_r^-}^{\Delta\phi_r^+} - \widehat{\Delta\phi}(k) \right) K_\phi(\hat{V}_a) \right]_{\Delta\phi_V^-}^{\Delta\phi_V^+} \cdot \\ & \quad \cdot K_{a-i} \frac{T_s}{\mathbf{z} - 1} \Big]_{l_a^-}^{l_a^+} \wedge \\ & \quad \left. \widehat{\Delta\delta}_{\hat{r}}(k) = K_r \frac{g_0}{\hat{V}_a(k)} \sin \hat{\phi}(k) + (drr(\hat{V}_a) - K_r) \hat{r}(k) \right\} \end{aligned}$$

Heading Hold FCL requirement

$$\begin{aligned} \mathcal{R}_{\widehat{HH}} = & \quad (B.47) \\ & \left\{ \left((\hat{\mathcal{U}}_p, \hat{\mathcal{X}}), \hat{\mathcal{U}}_c \right) \mid \forall k_1 \forall k_2 \left(1 \leq k_1 < k_2 < \infty \wedge \right. \right. \\ & \quad engaged(\widehat{SW}_{HH}(), k_1, k_2) \Rightarrow \\ & \quad \forall k \left(\widehat{\Delta\delta}_a(k) = \left[\left(\left[\widehat{\Delta\psi}(k_1) - \widehat{\Delta\psi}(k) \right] K_\psi(\hat{V}_a) \right]_{\Delta\phi_r^-}^{\Delta\phi_r^+} - \right. \right. \\ & \quad \left. \left. - \widehat{\Delta\phi}(k) \right) K_\phi(\hat{V}_a) \right]_{\Delta\phi_V^-}^{\Delta\phi_V^+} + dar \cdot \hat{r}(k) + \right. \\ & \quad \left. \left. \widehat{\Delta\delta}_{\hat{r}}(k) = K_r \frac{g_0}{\hat{V}_a(k)} \sin \hat{\phi}(k) + (drr(\hat{V}_a) - K_r) \hat{r}(k) \right) \right\} \end{aligned}$$

$$\begin{aligned}
& + \left[\left[\left(\widehat{\Delta\psi}(k_1) - \widehat{\Delta\psi}(k) \right) K_\psi(\hat{V}_a) \right]_{\Delta\phi_r^-}^{\Delta\phi_r^+} - \widehat{\Delta\phi}(k) \right) K_\phi(\hat{V}_a) \right]_{\Delta\phi_V^-}^{\Delta\phi_V^+} \cdot \\
& \cdot K_{a-i} \frac{T_s}{\mathbf{z} - 1} \Big]_{l_a^-}^{l_a^+} \wedge \\
& \widehat{\Delta\delta_{\hat{r}}}(k) = K_r \frac{g_0}{\hat{V}_a(k)} \sin \hat{\phi}(k) + (drr(\hat{V}_a) - K_r) \hat{r}(k) \Big) \Big\}
\end{aligned}$$

B.3 Fault modes

B.3.1 Control-surface fault modes

Partial loss of rudder surface

$$\mathcal{R}_{csf,1} = \tag{B.48}$$

$$\left\{ \left((\mathcal{U}_c, \mathcal{X}), \mathcal{C}_f \right) \middle| \forall t \exists C_{X_{\delta r}}() \exists C_{X_{\delta f}}() \exists C_{X_{\alpha \delta f}}() \exists C_{Y_{\delta a}}() \exists C_{Y_{\delta r}}() \exists C_{Y_{\alpha \delta r}}() \right. \\ \exists C_{Z_{\delta e}}() \exists C_{Z_{\beta^2 \delta e}}() \exists C_{Z_{\delta f}}() \exists C_{Z_{\alpha \delta f}}() \exists loss \left(\right. \\ 0 \leq t < \infty \wedge 0 < loss \leq 1 \wedge \\ C_{X_{\delta r}}(1 - \Delta C_{X_{\delta r}}\%) \leq C_{X_{\delta r}}(t) \leq C_{X_{\delta r}}(1 + \Delta C_{X_{\delta r}}\%) \wedge \\ C_{X_{\delta f}}(1 - \Delta C_{X_{\delta f}}\%) \leq C_{X_{\delta f}}(t) \leq C_{X_{\delta f}}(1 + \Delta C_{X_{\delta f}}\%) \wedge \\ C_{X_{\alpha \delta f}}(1 - \Delta C_{X_{\alpha \delta f}}\%) \leq C_{X_{\alpha \delta f}}(t) \leq C_{X_{\alpha \delta f}}(1 + \Delta C_{X_{\alpha \delta f}}\%) \wedge \\ C_{Y_{\delta a}}(1 - \Delta C_{Y_{\delta a}}\%) \leq C_{Y_{\delta a}}(t) \leq C_{Y_{\delta a}}(1 + \Delta C_{Y_{\delta a}}\%) \wedge \\ C_{Y_{\delta r}}(1 - \Delta C_{Y_{\delta r}}\%) \leq C_{Y_{\delta r}}(t) \leq C_{Y_{\delta r}}(1 + \Delta C_{Y_{\delta r}}\%) \wedge \\ C_{Y_{\alpha \delta r}}(1 - \Delta C_{Y_{\alpha \delta r}}\%) \leq C_{Y_{\alpha \delta r}}(t) \leq C_{Y_{\alpha \delta r}}(1 + \Delta C_{Y_{\alpha \delta r}}\%) \wedge \\ C_{Z_{\delta e}}(1 - \Delta C_{Z_{\delta e}}\%) \leq C_{Z_{\delta e}}(t) \leq C_{Z_{\delta e}}(1 + \Delta C_{Z_{\delta e}}\%) \wedge \\ C_{Z_{\beta^2 \delta e}}(1 - \Delta C_{Z_{\beta^2 \delta e}}\%) \leq C_{Z_{\beta^2 \delta e}}(t) \leq C_{Z_{\beta^2 \delta e}}(1 + \Delta C_{Z_{\beta^2 \delta e}}\%) \wedge \\ C_{Z_{\delta f}}(1 - \Delta C_{Z_{\delta f}}\%) \leq C_{Z_{\delta f}}(t) \leq C_{Z_{\delta f}}(1 + \Delta C_{Z_{\delta f}}\%) \wedge \\ C_{Z_{\alpha \delta f}}(1 - \Delta C_{Z_{\alpha \delta f}}\%) \leq C_{Z_{\alpha \delta f}}(t) \leq C_{Z_{\alpha \delta f}}(1 + \Delta C_{Z_{\alpha \delta f}}\%) \Rightarrow \\ X_{acd}(t) = q_{dyn}(t)S \left(loss \cdot C_{X_{\delta r}}(t)\delta_r(t) + C_{X_{\delta f}}(t)\delta_f(t) + C_{X_{\alpha \delta f}}(t)\alpha(t)\delta_f(t) \right) \wedge \\ Y_{acd}(t) = q_{dyn}(t)S \left(C_{Y_{\delta a}}(t)\delta_a(t) + loss \cdot C_{Y_{\delta r}}(t)\delta_r(t) + C_{Y_{\alpha \delta r}}(t)\alpha(t)\delta_r(t) \right) \wedge \\ \left. Z_{acd}(t) = q_{dyn}(t)S \left(C_{Z_{\delta e}}(t)\delta_e(t) + C_{Z_{\beta^2 \delta e}}(t)\beta^2(t)\delta_e(t) + C_{Z_{\delta f}}(t)\delta_f(t) \right) \right\}$$

$$\mathcal{R}_{csm,1} = \tag{B.49}$$

$$\left\{ \left((\mathcal{U}_c, \mathcal{X}), \mathcal{C}_m \right) \middle| \forall t \exists C_{l_{\delta a}}() \exists C_{l_{\delta r}}() \exists C_{l_{\alpha \delta a}}() \exists C_{m_{\delta e}}() C_{m_{\delta f}}() \right. \\ \exists C_{n_{\delta a}}() \exists C_{n_{\delta r}}() \exists loss \left(0 \leq t < \infty \wedge 0 < loss \leq 1 \wedge \right. \\ C_{l_{\delta a}}(1 - \Delta C_{l_{\delta a}}\%) \leq C_{l_{\delta a}}(t) \leq C_{l_{\delta a}}(1 + \Delta C_{l_{\delta a}}\%) \wedge \\ C_{l_{\delta r}}(1 - \Delta C_{l_{\delta r}}\%) \leq C_{l_{\delta r}}(t) \leq C_{l_{\delta r}}(1 + \Delta C_{l_{\delta r}}\%) \wedge \\ C_{l_{\alpha \delta a}}(1 - \Delta C_{l_{\alpha \delta a}}\%) \leq C_{l_{\alpha \delta a}}(t) \leq C_{l_{\alpha \delta a}}(1 + \Delta C_{l_{\alpha \delta a}}\%) \wedge \\ C_{m_{\delta e}}(1 - \Delta C_{m_{\delta e}}\%) \leq C_{m_{\delta e}}(t) \leq C_{m_{\delta e}}(1 + \Delta C_{m_{\delta e}}\%) \wedge \\ C_{m_{\delta f}}(1 - \Delta C_{m_{\delta f}}\%) \leq C_{m_{\delta f}}(t) \leq C_{m_{\delta f}}(1 + \Delta C_{m_{\delta f}}\%) \wedge \\ C_{n_{\delta a}}(1 - \Delta C_{n_{\delta a}}\%) \leq C_{n_{\delta a}}(t) \leq C_{n_{\delta a}}(1 + \Delta C_{n_{\delta a}}\%) \wedge \\ C_{n_{\delta r}}(1 - \Delta C_{n_{\delta r}}\%) \leq C_{n_{\delta r}}(t) \leq C_{n_{\delta r}}(1 + \Delta C_{n_{\delta r}}\%) \Rightarrow$$

$$\begin{aligned}
L_{acd}(t) &= q_{dyn}(t) S \frac{b}{2} (C_{l_{\delta a}}(t) \delta_a(t) + loss \cdot C_{l_{\delta r}}(t) \delta_r(t) + C_{l_{\alpha \delta a}}(t) \alpha(t) \delta_a(t)) \wedge \\
M_{acd}(t) &= q_{dyn}(t) S \bar{c} (C_{m_{\delta e}}(t) \delta_e(t) + C_{m_{\delta f}}(t) \delta_f(t)) \wedge \\
N_{acd}(t) &= q_{dyn}(t) S \frac{b}{2} (C_{n_{\delta a}}(t) \delta_a(t) + loss \cdot C_{n_{\delta r}}(t) \delta_r(t)) \Big) \Big\}
\end{aligned}$$

B.3.2 Engine fault modes

Engine loss

$$\begin{aligned}
\mathcal{R}_{pf,1} = & \tag{B.50} \\
& \left\{ \left((\mathcal{U}_c, \mathcal{X}), \mathcal{C}_f \right) \mid \forall t \left(0 \leq t < \infty \Rightarrow \right. \right. \\
& \left. \left. X_p(t) = 0 \wedge Y_p(t) = 0 \wedge Z_p(t) = 0 \right) \right\}
\end{aligned}$$

$$\begin{aligned}
\mathcal{R}_{pm,1} = & \tag{B.51} \\
& \left\{ \left((\mathcal{U}_c, \mathcal{X}), \mathcal{C}_m \right) \mid \forall t \left(0 \leq t < \infty \Rightarrow \right. \right. \\
& \left. \left. L_p(t) = 0 \wedge M_p(t) = 0 \wedge N_p(t) = 0 \right) \right\}
\end{aligned}$$

B.3.3 Actuator fault modes

Stuck rudder actuator

$$\begin{aligned}
\mathcal{R}_{rud,1} = & \tag{B.52} \\
& \left\{ \left((\mathcal{X}, \tilde{\mathcal{U}}_c), \mathcal{U}_c \right) \mid \exists \delta_r \forall t \left(\delta_r^- \leq \delta_r \leq \delta_r^+ \wedge 0 \leq t < \infty \Rightarrow \right. \right. \\
& \left. \left. \delta_r(t) = \delta_r \right) \right\}
\end{aligned}$$

Stuck aileron actuators

$$\begin{aligned}
\mathcal{R}_{ail,1} = & \tag{B.53} \\
& \left\{ \left((\mathcal{X}, \tilde{\mathcal{U}}_c), \mathcal{U}_c \right) \mid \exists \delta_a \forall t \left(\delta_a^- \leq \delta_a \leq \delta_a^+ \wedge 0 \leq t < \infty \Rightarrow \right. \right. \\
& \left. \left. \delta_a(t) = \delta_a \right) \right\}
\end{aligned}$$

Stuck elevator actuators

$$\mathcal{R}_{elv,1} = \left\{ \left((\mathcal{X}, \tilde{\mathcal{U}}_c), \mathcal{U}_c \right) \mid \exists \delta_e \forall t \left(\delta_e^- \leq \delta_e \leq \delta_e^+ \wedge 0 \leq t < \infty \Rightarrow \delta_e(t) = \delta_e \right) \right\} \quad (\text{B.54})$$

Stuck flap actuators

$$\mathcal{R}_{flp,1} = \left\{ \left((\mathcal{X}, \tilde{\mathcal{U}}_c), \mathcal{U}_c \right) \mid \exists \delta_f \forall t \left(\delta_f^- \leq \delta_f \leq \delta_f^+ \wedge 0 \leq t < \infty \Rightarrow \delta_f(t) = \delta_f \right) \right\} \quad (\text{B.55})$$

B.3.4 Rate gyro fault modes

Bad connection at roll rate gyro output

$$\mathcal{R}_{p,1} = \left\{ \left(\mathcal{X}, \tilde{\mathcal{X}} \right) \mid \forall t \left(0 \leq t < \infty \Rightarrow \tilde{p}(t) = OR_g^- \vee \tilde{p}(t) = OR_g^+ \right) \right\} \quad (\text{B.56})$$

Bad connection at pitch rate gyro output

$$\mathcal{R}_{q,1} = \left\{ \left(\mathcal{X}, \tilde{\mathcal{X}} \right) \mid \forall t \left(0 \leq t < \infty \Rightarrow \tilde{q}(t) = OR_g^- \vee \tilde{q}(t) = OR_g^+ \right) \right\} \quad (\text{B.57})$$

Bad connection at yaw rate gyro output

$$\mathcal{R}_{r,1} = \left\{ \left(\mathcal{X}, \tilde{\mathcal{X}} \right) \mid \forall t \left(0 \leq t < \infty \Rightarrow \tilde{r}(t) = OR_g^- \vee \tilde{r}(t) = OR_g^+ \right) \right\} \quad (\text{B.58})$$

B.3.5 Accelerometer fault modes

Bad connection at A_x accelerometer output

$$\mathcal{R}_{A_x,1} = \left\{ \left(\mathcal{X}, \tilde{\mathcal{X}} \right) \middle| \forall t \left(0 \leq t < \infty \Rightarrow \right. \right. \\ \left. \left. \tilde{A}_x(t) = OR_A^- \vee \tilde{A}_x(t) = OR_A^+ \right) \right\} \quad (\text{B.59})$$

Bad connection at A_y accelerometer output

$$\mathcal{R}_{A_y,1} = \left\{ \left(\mathcal{X}, \tilde{\mathcal{X}} \right) \middle| \forall t \left(0 \leq t < \infty \Rightarrow \right. \right. \\ \left. \left. \tilde{A}_y(t) = OR_A^- \vee \tilde{A}_y(t) = OR_A^+ \right) \right\} \quad (\text{B.60})$$

Bad connection at A_z accelerometer output

$$\mathcal{R}_{A_z,1} = \left\{ \left(\mathcal{X}, \tilde{\mathcal{X}} \right) \middle| \forall t \left(0 \leq t < \infty \Rightarrow \right. \right. \\ \left. \left. \tilde{A}_z(t) = OR_A^- \vee \tilde{A}_z(t) = OR_A^+ \right) \right\} \quad (\text{B.61})$$

B.3.6 Air data sensor fault modes

Bad connection at dynamic pressure sensor output

$$\mathcal{R}_{q_{dyn},1} = \left\{ \left(\mathcal{X}, \tilde{\mathcal{X}} \right) \middle| \forall t \left(0 \leq t < \infty \Rightarrow \right. \right. \\ \left. \left. \tilde{q}_{dyn}(t) = OR_{q_{dyn}}^- \vee \tilde{q}_{dyn}(t) = OR_{q_{dyn}}^+ \right) \right\} \quad (\text{B.62})$$

Bad connection at static pressure sensor output

$$\mathcal{R}_{p_s,1} = \left\{ \left(\mathcal{X}, \tilde{\mathcal{X}} \right) \mid \forall t \left(0 \leq t < \infty \Rightarrow \right. \right. \\ \left. \left. \tilde{p}_s(t) = OR_{p_s}^- \vee \tilde{p}_s(t) = OR_{p_s}^+ \right) \right\} \quad (\text{B.63})$$

Bad connection at temperature sensor output

$$\mathcal{R}_{T,1} = \left\{ \left(\mathcal{X}, \tilde{\mathcal{X}} \right) \mid \forall t \left(0 \leq t < \infty \Rightarrow \right. \right. \\ \left. \left. \tilde{T}(t) = OR_T^- \vee \tilde{T}(t) = OR_T^+ \right) \right\} \quad (\text{B.64})$$

B.3.7 Angle of attack sensor fault modes

Bad connection at angle of attack sensor output

$$\mathcal{R}_{\alpha,1} = \left\{ \left(\mathcal{X}, \tilde{\mathcal{X}} \right) \mid \forall t \left(0 \leq t < \infty \Rightarrow \right. \right. \\ \left. \left. \tilde{\alpha}(t) = OR_{\alpha}^- \vee \tilde{\alpha}(t) = OR_{\alpha}^+ \right) \right\} \quad (\text{B.65})$$

B.3.8 Attitude and heading sensor fault modes

Bad connection at pitch attitude sensor output

$$\mathcal{R}_{\theta,1} = \left\{ \left(\mathcal{X}, \tilde{\mathcal{X}} \right) \mid \forall t \left(0 \leq t < \infty \Rightarrow \right. \right. \\ \left. \left. \tilde{\theta}(t) = OR_{\phi/\theta}^- \vee \tilde{\theta}(t) = OR_{\phi/\theta}^+ \right) \right\} \quad (\text{B.66})$$

Bad connection at roll attitude sensor output

$$\mathcal{R}_{\phi,1} = \left\{ \left(\mathcal{X}, \tilde{\mathcal{X}} \right) \middle| \forall t \left(0 \leq t < \infty \Rightarrow \right. \right. \\ \left. \left. \tilde{\phi}(t) = OR_{\phi/\theta}^- \vee \tilde{\phi}(t) = OR_{\phi/\theta}^+ \right) \right\} \quad (\text{B.67})$$

Bad connection at heading sensor output

$$\mathcal{R}_{\psi,1} = \left\{ \left(\mathcal{X}, \tilde{\mathcal{X}} \right) \middle| \forall t \left(0 \leq t < \infty \Rightarrow \right. \right. \\ \left. \left. \tilde{\psi}(t) = OR_{\psi}^- \vee \tilde{\psi}(t) = OR_{\psi}^+ \right) \right\} \quad (\text{B.68})$$

B.4 DHC-2 requirement space restriction sets

Trim-condition restriction

$$\mathcal{S}_{C_{tr}} = \left\{ \mathcal{C}_{tr} \mid \exists a, \exists b \left(a = \cos(\alpha_{tr}) \cos(\beta_{tr}) \wedge \right. \right. \quad (B.69)$$

$$b = \sin(\phi_{tr}) \sin(\beta_{tr}) + \cos(\phi_{tr}) \sin(\alpha_{tr}) \cos(\beta_{tr}) \Rightarrow$$

$$\tan(\theta_{tr}) = \frac{a \cdot b + \sin(\gamma_{tr}) \sqrt{a^2 - \sin^2(\gamma_{tr}) + b^2}}{a^2 - \sin^2(\gamma_{tr})} \Big) \wedge$$

$$\left. trimCostFunction(\mathcal{C}_{tr}) < J_{tc}^+ \right\}$$

Autopilot operation restriction

$$\mathcal{S}_{ao} = \left\{ \mathcal{U}_p \mid \forall t \left(0 < t < \infty \Rightarrow \right. \quad (B.70)$$

$$\left(SW_{PAH}(t) = ON \vee SW_{ALH}(t) = ON \vee \right.$$

$$SW_{RAH}(t) = ON \vee SW_{HH}(t) = ON \Big) \wedge$$

$$\neg \left(SW_{PAH}(t) \wedge SW_{ALH}(t) \right) \wedge$$

$$\neg \left(SW_{RAH}(t) = ON \wedge SW_{HH}(t) = ON \right) \wedge$$

$$\neg \left(SW_{RAH}(t) = ON \wedge SW_{HS}(t) = ON \right) \wedge$$

$$\left. \neg \left(SW_{HH}(t) = ON \wedge SW_{HS}(t) = ON \right) \right\}$$

Atmospheric turbulence restriction

$$\mathcal{S}_{at} = \left\{ \left(\mathcal{U}_w, \mathcal{X} \right) \mid \forall t \left(0 < t < \infty \Rightarrow \right. \quad (B.71)$$

$$H(t) > H_{at} \wedge$$

$$\left(u_{wt}(t) = 0 \wedge w_{wt}(t) = 0 \wedge w_{wt}(t) = 0 \right) \vee$$

$$\exists \nu_{u_{wt}}() \exists \nu_{v_{wt}}() \exists \nu_{w_{wt}}() \left(\right.$$

$$whiteNoise(\nu_{u_{wt}}(), Npsd_{u_{wt}}) \wedge$$

$$whiteNoise(\nu_{v_{wt}}(), Npsd_{v_{wt}}) \wedge$$

$$\begin{aligned}
& whiteNoise(\nu_{w_{wt}}(), Npsd_{w_{wt}}) \Rightarrow \\
& u_{w_t}(t) \simeq h_{u_{wt}}(t, V_a(t)) \otimes \nu_{u_{wt}}(t) \wedge \\
& v_{w_t}(t) \simeq h_{v_{wt}}(t, V_a(t)) \otimes \nu_{v_{wt}}(t) \wedge \\
& w_{w_t}(t) \simeq h_{w_{wt}}(t, V_a(t)) \otimes \nu_{w_{wt}}(t) \wedge \\
& u_{\bar{w}}(t) = 0 \wedge v_{\bar{w}}(t) = 0 \wedge w_{\bar{w}}(t) = 0 \wedge \\
& u_{w_g}(t) = 0 \wedge v_{w_g}(t) = 0 \wedge w_{w_g}(t) = 0 \Big) \Big\} \tag{B.72}
\end{aligned}$$

B.5 Elementary requirements of interface blocks to AR-FTC system

FTC control panel requirement

$$\mathcal{R}_{FTC-CP} = \left\{ \left(\mathcal{U}_f, \tilde{\mathcal{U}}_f \right) \middle| \forall t \left(0 \leq t < \infty \Rightarrow \left(\widetilde{SW}_{FTC}(t) > V_{th} \Leftrightarrow SW_{FTC}(t) = ON \right) \right) \right\} \quad (B.73)$$

FTC-ADC card requirement

$$\mathcal{R}_{FTC-ADC} = \left\{ \left(\tilde{\mathcal{U}}_f, \bar{\mathcal{U}}_f \right) \middle| \forall k \left(0 \leq k < \infty \Rightarrow \left[S_{ADC} \left[\widetilde{SW}_{FTC}(k T_s) \right]_{ADC_{V-}}^{ADC_{V+}} + OFF_{ADC} \right] \right) \right\} \quad (B.74)$$

FTC input interface requirement

$$\mathcal{R}_{FTC-in} = \left\{ \left(\bar{\mathcal{U}}_f, \hat{\mathcal{U}}_f \right) \middle| \forall k \left(0 \leq k < \infty \Rightarrow \left(\widehat{SW}_{FTC}(k) = ON \Leftrightarrow (\bar{SW}_{FTC}(k) - OFF_{ADC}) S_{ADC}^{-1} > V_{th} \right) \right) \right\} \quad (B.75)$$

FTC safety switch requirement

$$\mathcal{R}_{FTC-SW} = \left\{ \left((\hat{\mathcal{U}}_f, \hat{\mathcal{X}}, \check{\mathcal{X}}), \hat{\mathcal{X}}' \right) \middle| \forall k \left(0 \leq k < \infty \Rightarrow \left(\widehat{SW}_{FTC}(k) = OFF \wedge \hat{\mathcal{X}}' = \hat{\mathcal{X}} \right) \vee \left(\widehat{SW}_{FTC}(k) = ON \wedge \hat{\mathcal{X}}' = \check{\mathcal{X}} \right) \right) \right\} \quad (B.76)$$

FTC output interface requirement

$$\begin{aligned}
 \mathcal{R}_{FTC-out} = & \quad (B.77) \\
 & \left\{ \left(\bar{\mathcal{Y}}_f, \tilde{\mathcal{Y}}_f \right) \mid \forall k \left(0 \leq k < \infty \Rightarrow \right. \right. \\
 & \quad \left(\bar{W}_p(k) > \left\lfloor S_{DAC} \left[V_{th} \right]_{DAC_{V-}}^{DAC_{V+}} + OFF_{DAC} \right\rfloor \Leftrightarrow \hat{W}_p(k) = ON \right) \wedge \\
 & \quad \left(\bar{W}_q(k) > \left\lfloor S_{DAC} \left[V_{th} \right]_{DAC_{V-}}^{DAC_{V+}} + OFF_{DAC} \right\rfloor \Leftrightarrow \hat{W}_q(k) = ON \right) \wedge \\
 & \quad \left(\bar{W}_r(k) > \left\lfloor S_{DAC} \left[V_{th} \right]_{DAC_{V-}}^{DAC_{V+}} + OFF_{DAC} \right\rfloor \Leftrightarrow \hat{W}_r(k) = ON \right) \wedge \\
 & \quad \left(\bar{W}_{ps}(k) > \left\lfloor S_{DAC} \left[V_{th} \right]_{DAC_{V-}}^{DAC_{V+}} + OFF_{DAC} \right\rfloor \Leftrightarrow \hat{W}_{ps}(k) = ON \right) \wedge \\
 & \quad \left(\bar{W}_{qdyn}(k) > \left\lfloor S_{DAC} \left[V_{th} \right]_{DAC_{V-}}^{DAC_{V+}} + OFF_{DAC} \right\rfloor \Leftrightarrow \hat{W}_{qdyn}(k) = ON \right) \wedge \\
 & \quad \left(\bar{W}_T(k) > \left\lfloor S_{DAC} \left[V_{th} \right]_{DAC_{V-}}^{DAC_{V+}} + OFF_{DAC} \right\rfloor \Leftrightarrow \hat{W}_T(k) = ON \right) \wedge \\
 & \quad \left(\bar{W}_\phi(k) > \left\lfloor S_{DAC} \left[V_{th} \right]_{DAC_{V-}}^{DAC_{V+}} + OFF_{DAC} \right\rfloor \Leftrightarrow \hat{W}_\phi(k) = ON \right) \wedge \\
 & \quad \left(\bar{W}_\theta(k) > \left\lfloor S_{DAC} \left[V_{th} \right]_{DAC_{V-}}^{DAC_{V+}} + OFF_{DAC} \right\rfloor \Leftrightarrow \hat{W}_\theta(k) = ON \right) \wedge \\
 & \quad \left. \left(\bar{W}_\psi(k) > \left\lfloor S_{DAC} \left[V_{th} \right]_{DAC_{V-}}^{DAC_{V+}} + OFF_{DAC} \right\rfloor \Leftrightarrow \hat{W}_\psi(k) = ON \right) \right\}
 \end{aligned}$$

FTC-DAC card requirement

$$\begin{aligned}
 \mathcal{R}_{FTC-DAC} = & \quad (B.78) \\
 & \left\{ \left(\bar{\mathcal{Y}}_f, \tilde{\mathcal{Y}}_f \right) \mid \forall k \forall t \left(\right. \right. \\
 & \quad 0 \leq k < \infty \wedge kT_s \leq t < (k+1)T_s \Rightarrow \\
 & \quad \tilde{W}_p(t) = (\bar{W}_p(k) - OFF_{DAC})S_{DAC}^{-1} \wedge \\
 & \quad \tilde{W}_q(t) = (\bar{W}_q(k) - OFF_{DAC})S_{DAC}^{-1} \wedge \\
 & \quad \tilde{W}_r(t) = (\bar{W}_r(k) - OFF_{DAC})S_{DAC}^{-1} \wedge \\
 & \quad \tilde{W}_{ps}(t) = (\bar{W}_{ps}(k) - OFF_{DAC})S_{DAC}^{-1} \wedge \\
 & \quad \tilde{W}_{qdyn}(t) = (\bar{W}_{qdyn}(k) - OFF_{DAC})S_{DAC}^{-1} \wedge \\
 & \quad \tilde{W}_T(t) = (\bar{W}_T(k) - OFF_{DAC})S_{DAC}^{-1} \wedge \\
 & \quad \tilde{W}_\phi(t) = (\bar{W}_\phi(k) - OFF_{DAC})S_{DAC}^{-1} \wedge \\
 & \quad \tilde{W}_\theta(t) = (\bar{W}_\theta(k) - OFF_{DAC})S_{DAC}^{-1} \wedge \\
 & \quad \left. \tilde{W}_\psi(t) = (\bar{W}_\psi(k) - OFF_{DAC})S_{DAC}^{-1} \right\}
 \end{aligned}$$

FTC display panel requirement

$$\begin{aligned}
 \mathcal{R}_{FTC-DP} = & \tag{B.79} \\
 & \left\{ \left(\mathcal{Y}_f, \mathcal{Y}_f \right) \middle| \forall k \left(0 \leq k < \infty \Rightarrow \right. \right. \\
 & \quad \tilde{W}_p(t) = ON \Leftrightarrow W_p(t) > V_{th} \wedge \\
 & \quad \tilde{W}_q(t) = ON \Leftrightarrow W_q(t) > V_{th} \wedge \\
 & \quad \tilde{W}_r(t) = ON \Leftrightarrow W_r(t) > V_{th} \wedge \\
 & \quad \tilde{W}_{p_s}(t) = ON \Leftrightarrow W_{p_s}(t) > V_{th} \wedge \\
 & \quad \tilde{W}_{q_{dyn}}(t) = ON \Leftrightarrow W_{q_{dyn}}(t) > V_{th} \wedge \\
 & \quad \tilde{W}_T(t) = ON \Leftrightarrow W_T(t) > V_{th} \wedge \\
 & \quad \tilde{W}_\phi(t) = ON \Leftrightarrow W_\phi(t) > V_{th} \wedge \\
 & \quad \tilde{W}_\psi(t) = ON \Leftrightarrow W_\psi(t) > V_{th} \wedge \\
 & \quad \left. \left. \tilde{W}_\theta(t) = ON \Leftrightarrow W_\theta(t) > V_{th} \right) \right\}
 \end{aligned}$$

Appendix C

Support tables of the specification

This appendix collects all tables supporting the relational specification of appendix B. The following list briefly describes the role and content of each table.

Table C.1 lists all elementary requirements, of the FTC environment, along with related domain and image variables, and expanded domain and image spaces. The table is divided in sub-tables according to the requirements partitioning performed in appendix B.

Table C.2 lists all requirements obtained by composition of elementary requirements of the FTC environment.

Table C.3 lists all fault-mode relations.

Table C.4 lists all pre-restriction sets.

Table C.5 lists all equivalent variables and the related equivalence classes that result from the first expansion over the domain and image spaces of the elementary specifications.

Table C.6 lists all domain and image variables used within the specification.

Table C.7 lists all constants along with their value, type, and description.

Table C.8 lists the quantified variables introduced in the relational specifications along with their type and description.

Table C.9 defines all auxiliary terms. The table is divided in sub-tables to distinguish between auxiliary terms, functions, and predicates.

Table C.10 defines all types used within the specification.

Table C.1: Elementary requirements

ID	Name	Domain variables	Image variables	Domain and image spaces	Extended domain and image spaces
AFCS performance requirements					
\mathcal{R}_{ALH}	altitude hold	$SW_{ALH}(), \phi(), V_a(),$ $u_{wt}(), v_{wt}(), w_{wt}(),$ $u_{wg}(), v_{wg}(), w_{wg}()$	$A_n(), H()$	$\mathcal{U}_w + \mathcal{U}_{p_a} + \mathcal{X} \times \mathcal{X}$	$\mathcal{U}_w + \mathcal{U}_{p_a} + \mathcal{X} + \mathcal{C}_{tr} \times \mathcal{X}$
\mathcal{R}_{HH}	heading hold	$SW_{HH}(),$ $u_{wt}(), v_{wt}(), w_{wt}(),$ $u_{wg}(), v_{wg}(), w_{wg}()$	$\phi(), \psi()$	$\mathcal{U}_w + \mathcal{U}_{p_a} \times \mathcal{X}$	$\mathcal{U}_w + \mathcal{U}_{p_a} + \mathcal{X} + \mathcal{C}_{tr} \times \mathcal{X}$
\mathcal{R}_{HS}	heading select	$SW_{HS}(), \psi_r(),$ $u_{wt}(), v_{wt}(), w_{wt}(),$ $u_{wg}(), v_{wg}(), w_{wg}()$	$\phi(), \psi(), p()$	$\mathcal{U}_w + \mathcal{U}_{p_a} \times \mathcal{X}$	$\mathcal{U}_w + \mathcal{U}_{p_a} + \mathcal{X} + \mathcal{C}_{tr} \times \mathcal{X}$
\mathcal{R}_{PAH}	pitch attitude hold	$SW_{PAH}(), \theta_r(), \phi(),$ $u_{wt}(), v_{wt}(), w_{wt}(),$ $u_{wg}(), v_{wg}(), w_{wg}()$	$\theta()$	$\mathcal{U}_w + \mathcal{U}_{p_a} + \mathcal{X} \times \mathcal{X}$	$\mathcal{U}_w + \mathcal{U}_{p_a} + \mathcal{X} + \mathcal{C}_{tr} \times \mathcal{X}$
\mathcal{R}_{RAH}	roll attitude hold	$SW_{RAH}(), \phi_r(),$ $u_{wt}(), v_{wt}(), w_{wt}(),$ $u_{wg}(), v_{wg}(), w_{wg}()$	$\phi()$	$\mathcal{U}_w + \mathcal{U}_{p_a} + \mathcal{X} \times \mathcal{X}$	$\mathcal{U}_w + \mathcal{U}_{p_a} + \mathcal{X} + \mathcal{C}_{tr} \times \mathcal{X}$
DHC-2 dynamics requirements					
\mathcal{R}_{Feq}	force equations of aircraft dynamics	$X_{a_{sd}}(), X_{a_{cd}}(),$ $X_p(), X_{gr}(), X_w(),$ $Y_{a_{sd}}(), Y_{a_{cd}}(),$ $Y_p(), Y_{gr}(), Y_w(),$ $Z_{a_{sd}}(), Z_{a_{cd}}(),$ $Z_p(), Z_{gr}(), Z_w(),$ $p(), q(), r()$	$V_a(), \alpha(), \beta()$	$\mathcal{C}_f + \mathcal{X} \times \mathcal{X}$	$\mathcal{C}_f + \mathcal{X} \times \mathcal{X}$
\mathcal{R}_{Keq}	kinematic equations	$p(), q(), r()$	$\psi(), \theta(), \phi()$	$\mathcal{X} \times \mathcal{X}$	$\mathcal{U}_c + \mathcal{U}_w + \mathcal{U}_{p_m} + \mathcal{X} \times \mathcal{X}$

Table C.1: Elementary requirements (continued)

ID	Name	Domain variables	Image variables	Domain and image spaces	Extended domain and image spaces
\mathcal{R}_{Meq}	moment equations of aircraft dynamics	$L_{a_{sd}}(), L_{a_{cd}}(), L_p(),$ $M_{a_{sd}}(), M_{a_{cd}}(), M_p(),$ $N_{a_{sd}}(), N_{a_{cd}}(), N_p()$	$p(), q(), r()$	$\mathcal{C}_m \times \mathcal{X}$	$\mathcal{C}_m \times \mathcal{X}$
\mathcal{R}_{Neq}	navigation equations	$V_a(), \alpha(), \beta(),$ $\theta(), \phi(),$ $u_{\bar{w}}(), v_{\bar{w}}(), w_{\bar{w}}(),$ $u_{wg}(), v_{wg}(), w_{wg}(),$ $u_{wt}(), v_{wt}(), w_{wt}()$	$H()$	$\mathcal{U}_w + \mathcal{X} \times \mathcal{X}$	$\mathcal{U}_c + \mathcal{U}_w + \mathcal{U}_{p_m} + \mathcal{X} \times \mathcal{X}$
\mathcal{R}_{ad}	air data	$H(), V_a()$	$T(), \rho(), p_s(), q_{dyn}()$	$\mathcal{X} \times \mathcal{X}$	$\mathcal{U}_c + \mathcal{U}_w + \mathcal{U}_{p_m} + \mathcal{X} \times \mathcal{X}$
\mathcal{R}_{aef}	aerodynamic forces (airframe)	$V_a(), \alpha(), \beta(),$ $p(), q(), r(), q_{dyn}()$	$X_{a_{sd}}(), Y_{a_{sd}}(), Z_{a_{sd}}()$	$\mathcal{X} \times \mathcal{C}_f$	$\mathcal{U}_c + \mathcal{U}_{p_m} + \mathcal{U}_w + \mathcal{X} \times \mathcal{C}_f + \mathcal{X}$
\mathcal{R}_{aem}	aerodynamic moments (airframe)	$V_a(), \alpha(), \beta(),$ $p(), q(), r(), q_{dyn}()$	$L_{a_{sd}}(), M_{a_{sd}}(), N_{a_{sd}}()$	$\mathcal{X} \times \mathcal{C}_m$	$\mathcal{U}_c + \mathcal{U}_{p_m} + \mathcal{U}_w + \mathcal{X} \times \mathcal{C}_m$
\mathcal{R}_{csf}	aerodynamic force (control surface)	$\alpha(), \beta(), q_{dyn}(),$ $\delta_a(), \delta_e(), \delta_r(), \delta_f()$	$X_{a_{cd}}(), Y_{a_{cd}}(), Z_{a_{cd}}()$	$\mathcal{U}_c + \mathcal{X} \times \mathcal{C}_f$	$\mathcal{U}_c + \mathcal{U}_{p_m} + \mathcal{U}_w + \mathcal{X} \times \mathcal{C}_f + \mathcal{X}$
\mathcal{R}_{csm}	aerodynamic moment (control surface)	$\alpha(), q_{dyn}(),$ $\delta_a(), \delta_e(), \delta_r(), \delta_f()$	$L_{a_{cd}}(), M_{a_{cd}}(), N_{a_{cd}}()$	$\mathcal{U}_c + \mathcal{X} \times \mathcal{C}_m$	$\mathcal{U}_c + \mathcal{U}_{p_m} + \mathcal{U}_w + \mathcal{X} \times \mathcal{C}_m$
\mathcal{R}_{grf}	gravity force	$\theta(), \phi()$	$X_{gr}(), Y_{gr}(), Z_{gr}()$	$\mathcal{X} \times \mathcal{C}_f$	$\mathcal{U}_c + \mathcal{U}_{p_m} + \mathcal{U}_w + \mathcal{X} \times \mathcal{C}_f + \mathcal{X}$

Table C.1: Elementary requirements (continued)

ID	Name	Domain variables	Image variables	Domain and image spaces	Extended domain and image spaces
\mathcal{R}_{ka}	kinematic acceleration (at crew station)	$V_a(), \alpha(), \beta(),$ $u_{\bar{w}}(), v_{\bar{w}}(), w_{\bar{w}}(),$ $u_{w_t}(), v_{w_t}(), w_{w_t}(),$ $u_{w_g}(), v_{w_g}(), w_{w_g}(),$ $p(), q(), r()$	$A_x(), A_y(), A_z()$	$\mathcal{U}_w + \mathcal{X} \times \mathcal{X}$	$\mathcal{U}_c + \mathcal{U}_w + \mathcal{U}_{p_m} + \mathcal{X} \times \mathcal{X}$
\mathcal{R}_{pf}	propulsive force	$q_{dyn}(), \rho(), V_a(), \alpha(),$ $n(), p_z()$	$X_p(), Y_p(), Z_p()$	$\mathcal{U}_{p_m} + \mathcal{X} \times \mathcal{C}_f$	$\mathcal{U}_c + \mathcal{U}_{p_m} + \mathcal{U}_w + \mathcal{X} \times \mathcal{C}_f + \mathcal{X}$
\mathcal{R}_{pm}	propulsive moment	$q_{dyn}(), \rho(), V_a(), \alpha(),$ $n(), p_z()$	$L_p(), M_p(), N_p()$	$\mathcal{U}_{p_m} + \mathcal{X} \times \mathcal{C}_m$	$\mathcal{U}_c + \mathcal{U}_{p_m} + \mathcal{U}_w + \mathcal{X} \times \mathcal{C}_m$
\mathcal{R}_{wf}	wind force	$u_{\bar{w}}(), v_{\bar{w}}(), w_{\bar{w}}(),$ $u_{w_t}(), v_{w_t}(), w_{w_t}(),$ $u_{w_g}(), v_{w_g}(), w_{w_g}(),$ $p(), q(), r()$	$X_w(), Y_w(), Z_w()$	$\mathcal{X} + \mathcal{U}_w \times \mathcal{C}_f$	$\mathcal{U}_c + \mathcal{U}_{p_m} + \mathcal{U}_w + \mathcal{X} \times \mathcal{C}_f + \mathcal{X}$
DHC-2 AFCS hardware requirements					
\mathcal{R}_{ADC}	A/D interface	$\tilde{p}(), \tilde{q}(), \tilde{r}(),$ $\tilde{A}_x(), \tilde{A}_y(), \tilde{A}_z(),$ $\tilde{p}_s(), \tilde{q}_{dyn}(), \tilde{\alpha}(),$ $\tilde{\psi}(), \tilde{\theta}(), \tilde{\phi}(),$ $\tilde{\psi}_r(), \tilde{\phi}_r(), \tilde{\theta}_r(),$ $\widetilde{SW}_{PAH}(), \widetilde{SW}_{ALH}(),$ $\widetilde{SW}_{RAH}(),$ $\widetilde{SW}_{HH}(), \widetilde{SW}_{HS}()$	$\bar{p}(), \bar{q}(), \bar{r}(),$ $\bar{A}_x(), \bar{A}_y(), \bar{A}_z(),$ $\bar{p}_s(), \bar{q}_{dyn}(), \bar{\alpha}(),$ $\bar{\psi}(), \bar{\theta}(), \bar{\phi}(),$ $\bar{\psi}_r(), \bar{\phi}_r(), \bar{\theta}_r(),$ $\bar{SW}_{PAH}(), \bar{SW}_{ALH}(),$ $\bar{SW}_{RAH}(),$ $\bar{SW}_{HH}(), \bar{SW}_{HS}()$	$\mathcal{X} + \mathcal{U}_{p_a} \times \mathcal{X} + \mathcal{U}_{p_a}$	$\mathcal{X} + \mathcal{U}_{p_a} \times \mathcal{X} + \mathcal{U}_{p_a}$
\mathcal{R}_{A_x}	A_x accelerometer	$A_x(), \theta()$	$\tilde{A}_x()$	$\mathcal{X} \times \tilde{\mathcal{X}}$	$\mathcal{U}_{p_a} + \mathcal{X} \times \mathcal{U}_{p_a} + \tilde{\mathcal{X}}$
\mathcal{R}_{A_y}	A_y accelerometer	$A_y(), \theta(), \phi()$	$\tilde{A}_y()$	$\mathcal{X} \times \tilde{\mathcal{X}}$	$\mathcal{U}_{p_a} + \mathcal{X} \times \mathcal{U}_{p_a} + \tilde{\mathcal{X}}$

Table C.1: Elementary requirements (continued)

ID	Name	Domain variables	Image variables	Domain and image spaces	Extended domain and image spaces
\mathcal{R}_{A_z}	A_z accelerometer	$A_z(), \theta(), \phi()$	$\tilde{A}_z()$	$\mathcal{X} \times \tilde{\mathcal{X}}$	$\mathcal{U}_{p_a} + \mathcal{X} \times \mathcal{U}_{p_a} + \tilde{\mathcal{X}}$
\mathcal{R}_{CP}	control panel	$\psi_r(), \theta_r(), \phi_r(), \dot{H}_r(),$ $SW_{PAH}(), SW_{ALH}(),$ $SW_{RAH}(),$ $SW_{HH}(), SW_{HS}()$	$\tilde{\psi}_r(), \tilde{\theta}_r(), \tilde{\phi}_r(), \tilde{\dot{H}}_r()$ $\widetilde{SW_{PAH}}(), \widetilde{SW_{ALH}}(),$ $\widetilde{SW_{RAH}}(),$ $\widetilde{SW_{HH}}(), \widetilde{SW_{HS}}()$	$\mathcal{U}_{p_a} \times \mathcal{U}_{p_a}$	$\mathcal{U}_{p_a} + \mathcal{X} \times \mathcal{U}_{p_a} + \tilde{\mathcal{X}}$
\mathcal{R}_{DAC}	D/A interface	$\bar{\delta}_e(), \bar{\delta}_a(), \bar{\delta}_r()$	$\tilde{\delta}_e(), \tilde{\delta}_a(), \tilde{\delta}_r()$	$\bar{\mathcal{U}}_c \times \tilde{\mathcal{U}}_c$	$\bar{\mathcal{U}}_c \times \mathcal{X} + \tilde{\mathcal{U}}_c$
\mathcal{R}_T	temperature sensor	$T()$	$\tilde{T}()$	$\mathcal{X} \times \tilde{\mathcal{X}}$	$\mathcal{U}_{p_a} + \mathcal{X} \times \mathcal{U}_{p_a} + \tilde{\mathcal{X}}$
\mathcal{R}_α	angle of attack sensor	$\alpha()$	$\tilde{\alpha}()$	$\mathcal{X} \times \tilde{\mathcal{X}}$	$\mathcal{U}_{p_a} + \mathcal{X} \times \mathcal{U}_{p_a} + \tilde{\mathcal{X}}$
\mathcal{R}_ϕ	roll attitude sensor	$\phi()$	$\tilde{\phi}()$	$\mathcal{X} \times \tilde{\mathcal{X}}$	$\mathcal{U}_{p_a} + \mathcal{X} \times \mathcal{U}_{p_a} + \tilde{\mathcal{X}}$
\mathcal{R}_ψ	heading sensor	$\psi()$	$\tilde{\psi}()$	$\mathcal{X} \times \tilde{\mathcal{X}}$	$\mathcal{U}_{p_a} + \mathcal{X} \times \mathcal{U}_{p_a} + \tilde{\mathcal{X}}$
\mathcal{R}_θ	pitch attitude sensor	$\theta()$	$\tilde{\theta}()$	$\mathcal{X} \times \tilde{\mathcal{X}}$	$\mathcal{U}_{p_a} + \mathcal{X} \times \mathcal{U}_{p_a} + \tilde{\mathcal{X}}$
\mathcal{R}_{ail}	aileron actuator	$\tilde{\delta}_a(), V_a(), p()$	$\delta_a()$	$\mathcal{X} + \tilde{\mathcal{U}}_c \times \mathcal{U}_c$	$\tilde{\mathcal{U}}_c + \mathcal{X} \times \mathcal{U}_c$
\mathcal{R}_{elv}	elevator actuator	$\tilde{\delta}_e(), V_a(), q()$	$\delta_e()$	$\mathcal{X} + \tilde{\mathcal{U}}_c \times \mathcal{U}_c$	$\tilde{\mathcal{U}}_c + \mathcal{X} \times \mathcal{U}_c$
\mathcal{R}_{p_s}	static pressure sensor	$p_s()$	$\tilde{p}_s()$	$\mathcal{X} \times \tilde{\mathcal{X}}$	$\mathcal{U}_{p_a} + \mathcal{X} \times \mathcal{U}_{p_a} + \tilde{\mathcal{X}}$
\mathcal{R}_p	roll rate gyro	$p()$	$\tilde{p}()$	$\mathcal{X} \times \tilde{\mathcal{X}}$	$\mathcal{U}_{p_a} + \mathcal{X} \times \mathcal{U}_{p_a} + \tilde{\mathcal{X}}$
$\mathcal{R}_{q_{dyn}}$	dynamic pressure sensor	$q_{dyn}()$	$\tilde{q}_{dyn}()$	$\mathcal{X} \times \tilde{\mathcal{X}}$	$\mathcal{U}_{p_a} + \mathcal{X} \times \mathcal{U}_{p_a} + \tilde{\mathcal{X}}$
\mathcal{R}_q	pitch rate gyro	$q()$	$\tilde{q}()$	$\mathcal{X} \times \tilde{\mathcal{X}}$	$\mathcal{U}_{p_a} + \mathcal{X} \times \mathcal{U}_{p_a} + \tilde{\mathcal{X}}$

Table C.1: Elementary requirements (continued)

ID	Name	Domain variables	Image variables	Domain and image spaces	Extended domain and image spaces
\mathcal{R}_{rud}	rudder actuator	$\tilde{\delta}_r(), V_a(), r()$	$\tilde{\delta}_r()$	$\mathcal{X} + \tilde{\mathcal{U}}_c \times \mathcal{U}_c$	$\tilde{\mathcal{U}}_c + \mathcal{X} \times \mathcal{U}_c$
\mathcal{R}_r	yaw rate gyro	$r()$	$\tilde{r}()$	$\mathcal{X} \times \tilde{\mathcal{X}}$	$\mathcal{U}_{p_a} + \mathcal{X} \times \tilde{\mathcal{U}}_{p_a} + \tilde{\mathcal{X}}$
DHC-2 AFCS software requirements					
\mathcal{R}_{in}	Software input interface	$\bar{p}(), \bar{q}(), \bar{r}(),$ $\bar{A}_x(), \bar{A}_y(), \bar{A}_z(),$ $\bar{p}_s(), \bar{q}_{dyn}(), \bar{\alpha}(),$ $\bar{\psi}(), \bar{\theta}(), \bar{\phi}(),$ $\bar{\psi}_r(), \bar{\phi}_r(), \bar{\theta}_r(),$ $\bar{SW}_{PAH}(), \bar{SW}_{ALH}(),$ $\bar{SW}_{RAH}(),$ $\bar{SW}_{HH}(), \bar{SW}_{HS}()$	$\hat{p}(), \hat{q}(), \hat{r}(),$ $\hat{A}_x(), \hat{A}_y(), \hat{A}_z(),$ $\hat{p}_s(), \hat{q}_{dyn}(), \hat{\alpha}(),$ $\hat{\psi}(), \hat{\theta}(), \hat{\phi}(),$ $\hat{\psi}_r(), \hat{\phi}_r(), \hat{\theta}_r(),$ $\widehat{SW}_{PAH}(), \widehat{SW}_{ALH}(),$ $\widehat{SW}_{RAH}(),$ $\widehat{SW}_{HH}(), \widehat{SW}_{HS}()$	$\mathcal{X} + \mathcal{U}_{p_a} \times \hat{\mathcal{X}} + \mathcal{U}_{p_a}$	$\mathcal{X} + \mathcal{U}_{p_a} \times \hat{\mathcal{X}} + \mathcal{U}_{p_a}$
\mathcal{R}_{out}	Software output interface	$\hat{\delta}_e(), \hat{\delta}_a(), \hat{\delta}_r()$	$\bar{\delta}_e(), \bar{\delta}_a(), \bar{\delta}_r()$	$\mathcal{U}_c \times \bar{\mathcal{U}}_c$	$\mathcal{U}_c \times \bar{\mathcal{U}}_c$
$\mathcal{R}_{\widehat{ALH}}$	altitude hold autopilot	$\hat{V}_a(), \hat{q}(), \hat{r}(), \hat{\phi}(), \hat{\theta}(),$ $\hat{H}(), \hat{H}_r(), \widehat{SW}_{ALH}()$	$\hat{\delta}_e()$	$\mathcal{U}_{p_a} + \hat{\mathcal{X}} \times \mathcal{U}_c$	$\mathcal{U}_{p_a} + \hat{\mathcal{X}} \times \mathcal{U}_c$
$\mathcal{R}_{\widehat{HH}}$	heading hold autopilot	$\hat{V}_a(), \hat{r}(), \hat{\phi}(), \hat{\psi}(),$ $\widehat{SW}_{HH}()$	$\hat{\delta}_a(), \hat{\delta}_r()$	$\mathcal{U}_{p_a} + \hat{\mathcal{X}} \times \mathcal{U}_c$	$\mathcal{U}_{p_a} + \hat{\mathcal{X}} \times \mathcal{U}_c$
$\mathcal{R}_{\widehat{HS}}$	heading select autopilot	$\hat{V}_a(), \hat{r}(), \hat{\phi}(), \hat{\psi}(),$ $\hat{\psi}_r(), \widehat{SW}_{HS}()$	$\hat{\delta}_a(), \hat{\delta}_r()$	$\mathcal{U}_{p_a} + \hat{\mathcal{X}} \times \mathcal{U}_c$	$\mathcal{U}_{p_a} + \hat{\mathcal{X}} \times \mathcal{U}_c$
$\mathcal{R}_{\widehat{PAH}}$	pitch attitude hold autopilot	$\hat{V}_a(), \hat{q}(), \hat{r}(), \hat{\phi}(), \hat{\theta}(),$ $\hat{\theta}_r(), \widehat{SW}_{PAH}()$	$\hat{\delta}_e()$	$\mathcal{U}_{p_a} + \hat{\mathcal{X}} \times \mathcal{U}_c$	$\mathcal{U}_{p_a} + \hat{\mathcal{X}} \times \mathcal{U}_c$
$\mathcal{R}_{\widehat{RAH}}$	roll attitude hold autopilot	$\hat{V}_a(), \hat{r}(), \hat{\phi}(),$ $\hat{\phi}_r(), \widehat{SW}_{RAH}()$	$\hat{\delta}_a(), \hat{\delta}_r()$	$\mathcal{U}_{p_a} + \hat{\mathcal{X}} \times \mathcal{U}_c$	$\mathcal{U}_{p_a} + \hat{\mathcal{X}} \times \mathcal{U}_c$
FTC interface requirements (the FTC acronym has been omitted from the identifier for the sake of space)					
\mathcal{R}_{-ADC}	FTC A/D card	$\widehat{SW}_{FTC}()$	$\bar{SW}_{FTC}()$	$\tilde{\mathcal{U}}_f \times \bar{\mathcal{U}}_f$	<i>not used</i>

Table C.1: Elementary requirements (continued)

ID	Name	Domain variables	Image variables	Domain and image spaces	Extended domain and image spaces
\mathcal{R}_{-CP}	FTC control panel	$SW_{FTC}()$	$\widetilde{SW}_{FTC}()$	$\mathcal{U}_f \times \tilde{\mathcal{U}}_f$	<i>not used</i>
\mathcal{R}_{-DAC}	FTC D/A card	$\bar{W}_p(), \bar{W}_q, \bar{W}_r,$ $\bar{W}_{ps}, \bar{W}_{qdyn}, \bar{W}_T,$ $\bar{W}_\phi, \bar{W}_\psi, \bar{W}_\theta$	$\tilde{W}_p(), \tilde{W}_q, \tilde{W}_r,$ $\tilde{W}_{ps}, \tilde{W}_{qdyn}, \tilde{W}_T,$ $\tilde{W}_\phi, \tilde{W}_\psi, \tilde{W}_\theta$	$\bar{\mathcal{Y}}_f \times \tilde{\mathcal{Y}}_f$	<i>not used</i>
\mathcal{R}_{-DP}	FTC display panel	$\tilde{W}_p(), \tilde{W}_q, \tilde{W}_r,$ $\tilde{W}_{ps}, \tilde{W}_{qdyn}, \tilde{W}_T,$ $\tilde{W}_\phi, \tilde{W}_\psi, \tilde{W}_\theta$	$W_p(), W_q, W_r,$ $W_{ps}, W_{qdyn}, W_T,$ W_ϕ, W_ψ, W_θ	$\tilde{\mathcal{Y}}_f \times \mathcal{Y}_f$	<i>not used</i>
\mathcal{R}_{-in}	FTC input interface	$\widehat{SW}_{FTC}()$	$\widehat{SW}_{FTC}()$	$\mathcal{U}_f \times \hat{\mathcal{U}}_f$	<i>not used</i>
\mathcal{R}_{-out}	FTC output interface	$\hat{W}_p(), \hat{W}_q, \hat{W}_r,$ $\hat{W}_{ps}, \hat{W}_{qdyn}, \hat{W}_T,$ $\hat{W}_\phi, \hat{W}_\psi, \hat{W}_\theta$	$\bar{W}_p(), \bar{W}_q, \bar{W}_r,$ $\bar{W}_{ps}, \bar{W}_{qdyn}, \bar{W}_T,$ $\bar{W}_\phi, \bar{W}_\psi, \bar{W}_\theta$	$\mathcal{Y}_f \times \bar{\mathcal{Y}}_f$	<i>not used</i>

Table C.2: Composed requirements

ID	Description	Extended domain and image spaces
AFCS performance specification		
\mathcal{R}_{PS}	performance specification	$\mathcal{C}_{tr} + \mathcal{U}_{pa} + \mathcal{X} + \mathcal{U}_w \times \mathcal{X}$
DHC-2 detail specification		
\mathcal{R}_{ACT}	control surface actuators	$\tilde{\mathcal{U}}_c + \mathcal{X} \times \mathcal{U}_c$
\mathcal{R}_{AFCS}	automatic flight control system	$\mathcal{U}_{pa} + \mathcal{X} \times \mathcal{U}_c$
\mathcal{R}_{Cin}	computer input	$\mathcal{U}_{pa} + \mathcal{X} \times \tilde{\mathcal{U}}_{pa} + \tilde{\mathcal{X}}$
\mathcal{R}_{DHC2}	DHC-2 airplane dynamics	$\mathcal{U}_c + \mathcal{U}_w + \mathcal{U}_{pm} + \mathcal{X} \times \mathcal{X}$
\mathcal{R}_{FCC}	flight control computer	$\tilde{\mathcal{U}}_{pa} + \tilde{\mathcal{X}} \times \mathcal{U}_c + \mathcal{X}$
\mathcal{R}_{FCL}	flight control laws	$\hat{\mathcal{U}}_{pa} + \hat{\mathcal{X}} \times \hat{\mathcal{U}}_c$
\mathcal{R}_{FCSw}	flight control software	$\tilde{\mathcal{U}}_{pa} + \tilde{\mathcal{X}} \times \mathcal{U}_c$
\mathcal{R}_{Sp}	primary sensors	$\mathcal{U}_{pa} + \mathcal{X} \times \tilde{\mathcal{U}}_{pa} + \tilde{\mathcal{X}}$
\mathcal{R}_{Ss}	secondary sensors	$\mathcal{U}_{pa} + \mathcal{X} \times \mathcal{U}_{pa} + \tilde{\mathcal{X}}$
\mathcal{R}_f	forces exerted upon the airplane	$\mathcal{U}_c + \mathcal{U}_{pm} + \mathcal{U}_w + \mathcal{X} \times \mathcal{C}_f + \mathcal{X}$
\mathcal{R}_m	moments exerted upon the airplane	$\mathcal{U}_c + \mathcal{U}_{pm} + \mathcal{U}_w + \mathcal{X} \times \mathcal{C}_m$

Table C.3: Fault modes

ID	Description	Domain and image spaces and variables
$\mathcal{R}_{csf,1}$	partial loss of rudder surface	same as \mathcal{R}_{csf}
$\mathcal{R}_{csm,1}$	partial loss of rudder surface	same as \mathcal{R}_{csm}
$\mathcal{R}_{pf,1}$	engine loss	same as \mathcal{R}_{pf}
$\mathcal{R}_{pm,1}$	engine loss	same as \mathcal{R}_{pm}
$\mathcal{R}_{A_x,1}$	Bad connection at A_x accelerometer output	same as \mathcal{R}_{A_x}
$\mathcal{R}_{A_y,1}$	Bad connection at A_y accelerometer output	same as \mathcal{R}_{A_y}
$\mathcal{R}_{A_z,1}$	Bad connection at A_z accelerometer output	same as \mathcal{R}_{A_z}
$\mathcal{R}_{T,1}$	Bad connection at temperature sensor output	same as \mathcal{R}_T
$\mathcal{R}_{\alpha,1}$	Bad connection at angle of attack sensor output	same as \mathcal{R}_{α}
$\mathcal{R}_{\phi,1}$	Bad connection at roll attitude sensor output	same as \mathcal{R}_{ϕ}
$\mathcal{R}_{\psi,1}$	Bad connection at heading sensor output	same as \mathcal{R}_{ψ}
$\mathcal{R}_{\theta,1}$	Bad connection at pitch attitude sensor output	same as \mathcal{R}_{θ}
$\mathcal{R}_{ail,1}$	Stuck aileron actuator	same as \mathcal{R}_{ail}
$\mathcal{R}_{elv,1}$	Stuck elevator actuator	same as \mathcal{R}_{elv}
$\mathcal{R}_{ps,1}$	Bad connection at static pressure sensor output	same as \mathcal{R}_{ps}
$\mathcal{R}_p,1$	Bad connection at roll rate gyro output	same as \mathcal{R}_p
\mathcal{R}_{qdyn}	Bad connection at dynamic pressure sensor output	same as \mathcal{R}_{qdyn}
$\mathcal{R}_q,1$	Bad connection at pitch rate gyro output	same as \mathcal{R}_q
$\mathcal{R}_{rud,1}$	Stuck rudder actuator	same as \mathcal{R}_{rud}
$\mathcal{R}_r,1$	Bad connection at yaw rate gyro output	same as \mathcal{R}_r

Table C.4: Restriction sets

ID	Description	Variables	Space	Extended space
$\mathcal{S}_{C_{tr}}$	Trim-condition	$V_{atr}, \alpha_{tr}, \beta_{tr}, p_{tr}, q_{tr}, r_{tr}, \psi_{tr}, \theta_{tr}, \phi_{tr}, H_{tr},$ $\gamma_{tr}, \delta_{atr}, \delta_{etr}, \delta_{rtr}, \delta_{ftr}, n_{tr}, p_{ztr}$	\mathcal{C}_{tr}	$\mathcal{C}_{tr} + \mathcal{U}_{p_a} + \mathcal{X} + \mathcal{U}_w$
\mathcal{S}_{ao}	Autopilot operation	$SW_{PAH}(), SW_{ALH}(),$ $SW_{RAH}(), SW_{HH}(), SW_{HS}()$	\mathcal{U}_{p_a}	$\mathcal{C}_{tr} + \mathcal{U}_{p_a} + \mathcal{X} + \mathcal{U}_w$
\mathcal{S}_{at}	Atmospheric turbulence	$u_{\bar{w}}(), v_{\bar{w}}(), w_{\bar{w}}(), u_{w_t}(), v_{w_t}(), w_{w_t}(),$ $u_{w_g}(), v_{w_g}(), w_{w_g}(), H()$	$\mathcal{U}_w + \mathcal{X}$	$\mathcal{C}_{tr} + \mathcal{U}_{p_a} + \mathcal{X} + \mathcal{U}_w$

Table C.5: Spaces used within the requirements specification

Space identifier	Space element identifier	Space element variables
Spaces introduced for the requirements specification of the FTC environment		
\mathcal{C}_f	\mathcal{C}_f	$X_{a_{sd}}(), Y_{a_{sd}}(), Z_{a_{sd}}(), X_{gr}(), Y_{gr}(), Z_{gr}(), X_w(), Y_w(), Z_w(), X_{a_{cd}}(), Y_{a_{cd}}(), Z_{a_{cd}}(), X_p(), Y_p(), Z_p()$
\mathcal{C}_m	\mathcal{C}_m	$L_{a_{sd}}(), M_{a_{sd}}(), N_{a_{sd}}(), L_{a_{cd}}(), M_{a_{cd}}(), N_{a_{cd}}(), L_p(), M_p(), N_p()$
\mathcal{C}_{tr}	\mathcal{C}_{tr}	$V_{atr}, \alpha_{tr}, \beta_{tr}, p_{tr}, q_{tr}, r_{tr}, \psi_{tr}, \theta_{tr}, \phi_{tr}, H_{tr}, \gamma_{tr}, \delta_{atr}, \delta_{etr}, \delta_{rtr}, \delta_{ftr}, n_{tr}, p_{ztr}$
\mathcal{U}_c	\mathcal{U}_c	$\delta_a(), \delta_e(), \delta_r()$
\mathcal{U}_{p_a}	\mathcal{U}_{p_a}	$SW_{PAH}(), SW_{ALH}(), SW_{RAH}(), SW_{HH}(), SW_{HS}(), \phi_r(), \psi_r(), \theta_r()$
\mathcal{U}_{p_m}	\mathcal{U}_{p_m}	$\delta_f(), n(), p_z()$
\mathcal{U}_w	\mathcal{U}_w	$u_{\bar{w}}(), v_{\bar{w}}(), w_{\bar{w}}(), u_{w_t}(), v_{w_t}(), w_{w_t}(), u_{w_g}(), v_{w_g}(), w_{w_g}()$
\mathcal{X}	\mathcal{X}	$V_a(), \alpha(), \beta(), p(), q(), r(), \psi(), \theta(), \phi(), H(), A_x(), A_y(), A_z(), p_s(), q_{dyn}(), T(), \rho()$
$\bar{\mathcal{U}}_c$	$\bar{\mathcal{U}}_c$	$\bar{\delta}_a(), \bar{\delta}_e(), \bar{\delta}_r()$
$\bar{\mathcal{U}}_{p_a}$	$\bar{\mathcal{U}}_{p_a}$	$\bar{SW}_{PAH}(), \bar{SW}_{ALH}(), \bar{SW}_{RAH}(), \bar{SW}_{HH}(), \bar{SW}_{HS}(), \bar{\phi}_r(), \bar{\psi}_r(), \bar{\theta}_r()$
$\bar{\mathcal{X}}$	$\bar{\mathcal{X}}$	$\bar{\alpha}(), \bar{\beta}(), \bar{p}(), \bar{q}(), \bar{r}(), \bar{\psi}(), \bar{\phi}(), \bar{\theta}(), \bar{A}_x(), \bar{A}_y(), \bar{A}_z(), \bar{p}_s(), \bar{q}_{dyn}(), \bar{T}()$
$\hat{\mathcal{U}}_c$	$\hat{\mathcal{U}}_c$	$\hat{\delta}_a(), \hat{\delta}_e(), \hat{\delta}_r()$
$\hat{\mathcal{U}}_{p_a}$	$\hat{\mathcal{U}}_{p_a}$	$\widehat{SW}_{PAH}(), \widehat{SW}_{ALH}(), \widehat{SW}_{RAH}(), \widehat{SW}_{HH}(), \widehat{SW}_{HS}(), \hat{\phi}_r(), \hat{\psi}_r(), \hat{\theta}_r()$
$\hat{\mathcal{X}}$	$\hat{\mathcal{X}}$	$\hat{V}_a(), \hat{\alpha}(), \hat{\beta}(), \hat{p}(), \hat{q}(), \hat{r}(), \hat{\psi}(), \hat{\theta}(), \hat{\phi}(), \hat{H}(), \hat{A}_x(), \hat{A}_y(), \hat{A}_z(), \hat{p}_s(), \hat{q}_{dyn}(), \hat{T}()$
$\tilde{\mathcal{U}}_c$	$\tilde{\mathcal{U}}_c$	$\tilde{\delta}_a(), \tilde{\delta}_e(), \tilde{\delta}_r()$
$\tilde{\mathcal{U}}_{p_a}$	$\tilde{\mathcal{U}}_{p_a}$	$\widetilde{SW}_{PAH}(), \widetilde{SW}_{ALH}(), \widetilde{SW}_{RAH}(), \widetilde{SW}_{HH}(), \widetilde{SW}_{HS}(), \tilde{\phi}_r(), \tilde{\psi}_r(), \tilde{\theta}_r()$
$\tilde{\mathcal{X}}$	$\tilde{\mathcal{X}}$	$\tilde{\alpha}(), \tilde{\beta}(), \tilde{p}(), \tilde{q}(), \tilde{r}(), \tilde{\psi}(), \tilde{\theta}(), \tilde{\phi}(), \tilde{A}_x(), \tilde{A}_y(), \tilde{A}_z(), \tilde{p}_s(), \tilde{q}_{dyn}(), \tilde{T}()$
Spaces introduced for the requirements specification of the FTC system		
\mathcal{U}_f	\mathcal{U}_f	$SW_{FTC}()$
\mathcal{Y}_f	\mathcal{Y}_f	$W_p(), W_q(), W_r(), W_{p_s}(), W_{q_{dyn}}(), W_T(), W_\phi(), W_\theta(), W_\psi()$
$\bar{\mathcal{U}}_f$	$\bar{\mathcal{U}}_f$	$\bar{SW}_{FTC}()$
$\bar{\mathcal{Y}}_f$	$\bar{\mathcal{Y}}_f$	$\bar{W}_p(), \bar{W}_q(), \bar{W}_r(), \bar{W}_{p_s}(), \bar{W}_{q_{dyn}}(), \bar{W}_T(), \bar{W}_\phi(), \bar{W}_\theta(), \bar{W}_\psi()$
$\hat{\mathcal{U}}_f$	$\hat{\mathcal{U}}_f$	$\hat{SW}_{FTC}()$
$\hat{\mathcal{Y}}_f$	$\hat{\mathcal{Y}}_f$	$\hat{W}_p(), \hat{W}_q(), \hat{W}_r(), \hat{W}_{p_s}(), \hat{W}_{q_{dyn}}(), \hat{W}_T(), \hat{W}_\phi(), \hat{W}_\theta(), \hat{W}_\psi()$

Table C.5: Spaces used within the requirements specification (continued)

Space identifier	Space element identifier	Space element variables
$\tilde{\mathcal{U}}_f$ $\tilde{\mathcal{Y}}_f$	$\tilde{\mathcal{U}}_f$ $\tilde{\mathcal{Y}}_f$	$S\tilde{W}_{FTC}()$ $\tilde{W}_p(), \tilde{W}_q(), \tilde{W}_r(), \tilde{W}_{ps}(), \tilde{W}_{qdyn}(), \tilde{W}_T(), \tilde{W}_\phi(), \tilde{W}_\theta(), \tilde{W}_\psi()$

Table C.6: Domain and image variables

<i>ID</i>	Type	Description
Actual-quantity variables		
$A_x()$	time-T \rightarrow acceleration-T	component along X_B -axis of kinematic acceleration at crew station
$A_y()$	time-T \rightarrow acceleration-T	component along Y_B -axis of kinematic acceleration at crew station
$A_z()$	time-T \rightarrow acceleration-T	component along Z_B -axis of kinematic acceleration at crew station
$H()$	time-T \rightarrow altitude-T	height above sea level
$L_p()$	time-T \rightarrow moment-T	propulsive moment about X_B -axis
$L_{acd}()$	time-T \rightarrow moment-T	control-surface-exerted aerodynamic moment about X_B -axis
$L_{asd}()$	time-T \rightarrow moment-T	airframe-exerted aerodynamic moment about X_B -axis
$M_p()$	time-T \rightarrow moment-T	propulsive moment about Y_B -axis
$M_{acd}()$	time-T \rightarrow moment-T	control-surface-exerted aerodynamic moment about Y_B -axis
$M_{asd}()$	time-T \rightarrow moment-T	airframe-exerted aerodynamic moment about Y_B -axis
$N_p()$	time-T \rightarrow moment-T	propulsive moment about Z_B -axis
$N_{acd}()$	time-T \rightarrow moment-T	control-surface-exerted aerodynamic moment about Z_B -axis
$N_{asd}()$	time-T \rightarrow moment-T	airframe-exerted aerodynamic moment about Z_B -axis
$SW_{ALH}()$	time-T \rightarrow switch-T	ALH on/off switch
$SW_{HH}()$	time-T \rightarrow switch-T	HH on/off switch
$SW_{HS}()$	time-T \rightarrow switch-T	HS on/off switch
$SW_{PAH}()$	time-T \rightarrow switch-T	PAH on/off switch
$SW_{RAH}()$	time-T \rightarrow switch-T	RAH on/off switch
$T()$	time-T \rightarrow temperature-T	air temperature
$V_a()$	time-T \rightarrow airspeed-T	airspeed
$X_p()$	time-T \rightarrow force-T	propulsive force along X_B -axis
$X_w()$	time-T \rightarrow force-T	wind force along X_B -axis
$X_{acd}()$	time-T \rightarrow force-T	control-surface-exerted aerodynamic force along X_B -axis
$X_{asd}()$	time-T \rightarrow force-T	airframe-exerted aerodynamic force along X_B -axis
$X_{gr}()$	time-T \rightarrow force-T	gravity force along X_B -axis
$Y_p()$	time-T \rightarrow force-T	propulsive force along Y_B -axis
$Y_w()$	time-T \rightarrow force-T	wind force along Y_B -axis

Table C.6: Domain and image variables (continued)

<i>ID</i>	Type	Description
$Y_{acd}()$	time-T \rightarrow force-T	control-surface-exerted aerodynamic force along Y_B -axis
$Y_{asd}()$	time-T \rightarrow force-T	airframe-exerted aerodynamic force along Y_B -axis
$Y_{gr}()$	time-T \rightarrow force-T	gravity force along Y_B -axis
$Z_p()$	time-T \rightarrow force-T	propulsive force along Z_B -axis
$Z_w()$	time-T \rightarrow force-T	wind force along Z_B -axis
$Z_{acd}()$	time-T \rightarrow force-T	control-surface-exerted aerodynamic force along Z_B -axis
$Z_{asd}()$	time-T \rightarrow force-T	airframe-exerted aerodynamic force along Z_B -axis
$Z_{gr}()$	time-T \rightarrow force-T	gravity force along Z_B -axis
$\alpha()$	time-T \rightarrow AOT-T	angle of attack
$\beta()$	time-T \rightarrow angle-T	sideslip angle
$\delta_a()$	time-T \rightarrow aileronDeflection-T	aileron deflection ($\delta_a = \delta_{a_{right}} - \delta_{a_{left}}$)
$\delta_e()$	time-T \rightarrow elevatorDeflection-T	elevator deflection
$\delta_f()$	time-T \rightarrow flapDeflection-T	flap deflection
$\delta_r()$	time-T \rightarrow rudderDeflection-T	rudder deflection
$\phi()$	time-T \rightarrow angle-T	roll angle
$\phi_r()$	time-T \rightarrow bankReference-T	reference roll angle
$\psi()$	time-T \rightarrow angle-T	heading angle
$\psi_r()$	time-T \rightarrow headingReference-T	reference heading angle
$\rho()$	time-T \rightarrow density-T	air density
$\theta()$	time-T \rightarrow angle-T	pitch angle
$\theta_r()$	time-T \rightarrow pitchReference-T	reference pitch angle
$n()$	time-T \rightarrow engineSpeed-T	engine speed
$p()$	time-T \rightarrow angularVelocity-T	roll angular rate
$p_s()$	time-T \rightarrow pressure-T	static pressure
$p_z()$	time-T \rightarrow pressure-T	engine manifold pressure
$q()$	time-T \rightarrow angularVelocity-T	pitch angular rate
$q_{dyn}()$	time-T \rightarrow pressure-T	dynamic pressure
$r()$	time-T \rightarrow angularVelocity-T	yaw angular rate

Table C.6: Domain and image variables (continued)

ID	Type	Description
$u_{\bar{w}}()$	time-T \rightarrow velocity-T	mean wind velocity component along X_B -axis
$u_{w_g}()$	time-T \rightarrow velocity-T	wind-gust velocity component along X_B -axis
$u_{w_t}()$	time-T \rightarrow velocity-T	wind-turbulence velocity component along X_B -axis
$v_{\bar{w}}()$	time-T \rightarrow velocity-T	mean wind velocity component along Y_B -axis
$v_{w_g}()$	time-T \rightarrow velocity-T	wind-gust velocity component along Y_B -axis
$v_{w_t}()$	time-T \rightarrow velocity-T	wind-turbulence velocity component along Y_B -axis
$w_{\bar{w}}()$	time-T \rightarrow velocity-T	mean wind velocity component along Z_B -axis
$w_{w_g}()$	time-T \rightarrow velocity-T	wind-gust velocity component along Z_B -axis
$w_{w_t}()$	time-T \rightarrow velocity-T	wind-turbulence velocity component along Z_B -axis
Software-quantity variables		
$\bar{A}_x()$	natural-T \rightarrow ADCoutput-T	ADC software representation of $\tilde{A}_x()$
$\bar{A}_y()$	natural-T \rightarrow ADCoutput-T	ADC software representation of $\tilde{A}_y()$
$\bar{A}_z()$	natural-T \rightarrow ADCoutput-T	ADC software representation of $\tilde{A}_z()$
$\bar{S\bar{W}}_{ALH}()$	natural-T \rightarrow ADCoutput-T	ADC software representation of $\widetilde{S\bar{W}}_{ALH}()$
$\bar{S\bar{W}}_{HH}()$	natural-T \rightarrow ADCoutput-T	ADC software representation of $\widetilde{S\bar{W}}_{HH}()$
$\bar{S\bar{W}}_{HS}()$	natural-T \rightarrow ADCoutput-T	ADC software representation of $\widetilde{S\bar{W}}_{HS}()$
$\bar{S\bar{W}}_{PAH}()$	natural-T \rightarrow ADCoutput-T	ADC software representation of $\widetilde{S\bar{W}}_{PAH}()$
$\bar{S\bar{W}}_{RAH}()$	natural-T \rightarrow ADCoutput-T	ADC software representation of $\widetilde{S\bar{W}}_{RAH}()$
$\bar{T}()$	natural-T \rightarrow ADCoutput-T	ADC software representation of $\tilde{T}()$
$\bar{\alpha}()$	natural-T \rightarrow ADCoutput-T	ADC software representation of $\tilde{\alpha}()$
$\bar{\delta}_a()$	natural-T \rightarrow DACinput-T	DAC software representation of $\tilde{\delta}_a()$
$\bar{\delta}_e()$	natural-T \rightarrow DACinput-T	DAC software representation of $\tilde{\delta}_e()$
$\bar{\delta}_r()$	natural-T \rightarrow DACinput-T	DAC software representation of $\tilde{\delta}_r()$
$\bar{\phi}()$	natural-T \rightarrow ADCoutput-T	ADC software representation of $\tilde{\phi}()$
$\bar{\phi}_r()$	natural-T \rightarrow ADCoutput-T	ADC software representation of $\tilde{phi}_r()$
$\bar{\psi}()$	natural-T \rightarrow ADCoutput-T	ADC software representation of $\tilde{\psi}()$
$\bar{\psi}_r()$	natural-T \rightarrow ADCoutput-T	ADC software representation of $\tilde{\psi}_r()$

Table C.6: Domain and image variables (continued)

<i>ID</i>	Type	Description
$\bar{\theta}()$	natural-T \rightarrow ADCoutput-T	ADC software representation of $\bar{\theta}()$
$\bar{\theta}_r()$	natural-T \rightarrow ADCoutput-T	ADC software representation of $\bar{\theta}_r()$
$\bar{p}()$	natural-T \rightarrow ADCoutput-T	ADC software representation of $\bar{p}()$
$\bar{p}_s()$	natural-T \rightarrow ADCoutput-T	ADC software representation of $\bar{p}_s()$
$\bar{q}()$	natural-T \rightarrow ADCoutput-T	ADC software representation of $\bar{q}()$
$\bar{q}_{dyn}()$	natural-T \rightarrow ADCoutput-T	ADC software representation of $\bar{q}_{dyn}()$
$\bar{r}()$	natural-T \rightarrow ADCoutput-T	ADC software representation of $\bar{r}()$
$\hat{A}_x()$	natural-T \rightarrow acceleration-T	software variable representing $A_x()$
$\hat{A}_y()$	natural-T \rightarrow acceleration-T	software variable representing $A_y()$
$\hat{A}_z()$	natural-T \rightarrow acceleration-T	software variable representing $A_z()$
$\hat{H}()$	natural-T \rightarrow altitude-T	software variable representing $H()$
$\widehat{SW}_{ALH}()$	natural-T \rightarrow switch-T	software variable representing $SW_{ALH}()$
$\widehat{SW}_{HH}()$	natural-T \rightarrow switch-T	software variable representing $SW_{HH}()$
$\widehat{SW}_{HS}()$	natural-T \rightarrow switch-T	software variable representing $SW_{HS}()$
$\widehat{SW}_{PAH}()$	natural-T \rightarrow switch-T	software variable representing $SW_{PAH}()$
$\widehat{SW}_{RAH}()$	natural-T \rightarrow switch-T	software variable representing $SW_{RAH}()$
$\hat{V}_a()$	natural-T \rightarrow airspeed-T	software variable representing $V_a()$
$\hat{\alpha}()$	natural-T \rightarrow angle-T	software variable representing $\alpha()$
$\hat{\delta}_a()$	natural-T \rightarrow aileronDeflection-T	software variable representing $\delta_a()$
$\hat{\delta}_e()$	natural-T \rightarrow elevatorDeflection-T	software variable representing $\delta_e()$
$\hat{\delta}_r()$	natural-T \rightarrow rudderDeflection-T	software variable representing $\delta_r()$
$\hat{\phi}()$	natural-T \rightarrow angle-T	software variable representing $\phi()$
$\hat{\phi}_r()$	natural-T \rightarrow bankReference-T	software variable representing $\phi_r()$
$\hat{\psi}()$	natural-T \rightarrow angle-T	software variable representing $\psi()$
$\hat{\psi}_r()$	natural-T \rightarrow headingReference-T	software variable representing $\psi_r()$
$\hat{\theta}()$	natural-T \rightarrow angle-T	software variable representing $\theta()$
$\hat{p}()$	natural-T \rightarrow angularVelocity-T	software variable representing $p()$

Table C.6: Domain and image variables (continued)

<i>ID</i>	Type	Description
$\hat{p}_s()$	natural-T \rightarrow pressure-T	software variable representing $p_s()$
$\hat{q}()$	natural-T \rightarrow angularVelocity-T	software variable representing $q()$
$\hat{q}_{dyn}()$	natural-T \rightarrow pressure-T	software variable representing $q_{dyn}()$
$\hat{r}()$	natural-T \rightarrow angularVelocity-T	software variable representing $r()$
Electrical-quantity variables		
$\tilde{A}_x()$	time-T \rightarrow accelerometerOutput-T	output of accelerometer along X_B -axis
$\tilde{A}_y()$	time-T \rightarrow accelerometerOutput-T	output of accelerometer along Y_B -axis
$\tilde{A}_z()$	time-T \rightarrow accelerometerOutput-T	output of accelerometer along Z_B -axis
$\widetilde{SW}_{ALH}()$	time-T \rightarrow CPOutput-T	output of ALH control switch
$\widetilde{SW}_{HH}()$	time-T \rightarrow CPOutput-T	output of HH control switch
$\widetilde{SW}_{HS}()$	time-T \rightarrow CPOutput-T	output of HS control switch
$\widetilde{SW}_{PAH}()$	time-T \rightarrow CPOutput-T	output of PAH control switch
$\widetilde{SW}_{RAH}()$	time-T \rightarrow CPOutput-T	output of RAH control switch
$\tilde{T}()$	time-T \rightarrow TSensorOutput-T	output of temperature sensor
$\tilde{\alpha}()$	time-T \rightarrow AOTSensorOutput-T	output of angle of attack sensor
$\tilde{\delta}_a()$	time-T \rightarrow PCUInput-T	input to aileron PCU (+0.5 $\tilde{\alpha}()$ to right aileron PCU and $(-0.5\tilde{\alpha}()$ to left aileron PCU
$\tilde{\delta}_e()$	time-T \rightarrow PCUInput-T	input to elevator PCU
$\tilde{\delta}_f()$	time-T \rightarrow PCUInput-T	input to flap PCU
$\tilde{\delta}_r()$	time-T \rightarrow PCUInput-T	input to rudder PCU
$\tilde{\phi}()$	time-T \rightarrow attitudeSensorOuput-T	output of roll attitude sensor
$\tilde{\phi}_r()$	time-T \rightarrow CPOutput-T	output of reference roll control knob
$\tilde{\psi}()$	time-T \rightarrow headingSensorOutput-T	output of heading sensor
$\tilde{\psi}_r()$	time-T \rightarrow CPOutput-T	output of reference heading control knob
$\tilde{\theta}()$	time-T \rightarrow attitudeSensorOuput-T	output of pitch attitude sensor
$\tilde{\theta}_r()$	time-T \rightarrow attitudeSensorOuput-T	output of reference pitch control knob
$\tilde{p}()$	time-T \rightarrow gyroOutput-T	output of roll rate gyro

Table C.6: Domain and image variables (continued)

<i>ID</i>	Type	Description
$\tilde{p}_s()$	time-T \rightarrow psSensorOutput-T	output of static pressure sensor
$\tilde{q}()$	time-T \rightarrow gyroOutput-T	output of pitch rate gyro
$\tilde{q}_{dyn}()$	time-T \rightarrow qdynSensorOutput-T	output of dynamic pressure sensor
$\tilde{r}()$	time-T \rightarrow gyroOutput-T	output of yaw rate gyro
Actual-quantity variables (FTC system)		
$SW_{FTC}()$	time-T \rightarrow switch-T	FTC on/off switch
$W_T()$	time-T \rightarrow warningLight-T	temperature sensor warning light
$W_\phi()$	time-T \rightarrow warningLight-T	roll attitude sensor warning light
$W_\psi()$	time-T \rightarrow warningLight-T	heading sensor warning light
$W_\theta()$	time-T \rightarrow warningLight-T	pitch attitude sensor warning light
$W_p()$	time-T \rightarrow warningLight-T	roll rate gyro warning light
$W_q()$	time-T \rightarrow warningLight-T	pitch rate gyro warning light
$W_r()$	time-T \rightarrow warningLight-T	yaw rate gyro warning light
$W_{p_s}()$	time-T \rightarrow warningLight-T	static pressure sensor warning light
$W_{q_{dyn}}()$	time-T \rightarrow warningLight-T	dynamic pressure sensor warning light
Software-quantity variables (FTC system)		
$\bar{S}W_{FTC}()$	natural-T \rightarrow DACinput-T	DAC software representation of $SW_{FTC}()$
$\bar{W}_T()$	natural-T \rightarrow DACinput-T	DAC software representation of $T()$
$\bar{W}_\phi()$	natural-T \rightarrow DACinput-T	DAC software representation of $\phi()$
$\bar{W}_\psi()$	natural-T \rightarrow DACinput-T	DAC software representation of $\psi()$
$\bar{W}_\theta()$	natural-T \rightarrow DACinput-T	DAC software representation of $\theta()$
$\bar{W}_p()$	natural-T \rightarrow DACinput-T	DAC software representation of $p()$
$\bar{W}_q()$	natural-T \rightarrow DACinput-T	DAC software representation of $q()$
$\bar{W}_r()$	natural-T \rightarrow DACinput-T	DAC software representation of $r()$
$\bar{W}_{p_s}()$	natural-T \rightarrow DACinput-T	DAC software representation of $p_s()$
$\bar{W}_{q_{dyn}}()$	natural-T \rightarrow DACinput-T	DAC software representation of $q_{dyn}()$
$\widehat{SW}_{FTC}()$	natural-T \rightarrow switch-T	software variable representing $SW_{FTC}()$

Table C.6: Domain and image variables (continued)

ID	Type	Description
$\hat{W}_T()$	natural-T \rightarrow warningLight-T	software variable representing $W_T()$
$\hat{W}_\phi()$	natural-T \rightarrow warningLight-T	software variable representing $W_\phi()$
$\hat{W}_\psi()$	natural-T \rightarrow warningLight-T	software variable representing $W_\psi()$
$\hat{W}_\theta()$	natural-T \rightarrow warningLight-T	software variable representing $W_\theta()$
$\hat{W}_p()$	natural-T \rightarrow warningLight-T	software variable representing $W_p()$
$\hat{W}_q()$	natural-T \rightarrow warningLight-T	software variable representing $W_q()$
$\hat{W}_r()$	natural-T \rightarrow warningLight-T	software variable representing $W_r()$
$\hat{W}_{p_s}()$	natural-T \rightarrow warningLight-T	software variable representing $W_{p_s}()$
$\hat{W}_{q_{dyn}}()$	natural-T \rightarrow warningLight-T	software variable representing $W_{q_{dyn}}()$
Electrical-quantity variables (FTC system)		
$\tilde{S}W_{FTC}()$	time-T \rightarrow CPOutput-T	output of FTC control switch
$\tilde{W}_p()$	time-T \rightarrow warningLightInput-T	input to roll rate gyro warning light
$\tilde{W}_q()$	time-T \rightarrow warningLightInput-T	input to pitch rate gyro warning light
$\tilde{W}_r()$	time-T \rightarrow warningLightInput-T	input to yaw rate gyro warning light
$\tilde{W}_{p_s}()$	time-T \rightarrow warningLightInput-T	input to static pressure sensor warning light
$\tilde{W}_{q_{dyn}}()$	time-T \rightarrow warningLightInput-T	input to dynamic pressure sensor warning light
$\tilde{W}_T()$	time-T \rightarrow warningLightInput-T	input to temperature sensor warning light
$\tilde{W}_\phi()$	time-T \rightarrow warningLightInput-T	input to roll attitude sensor warning light
$\tilde{W}_\theta()$	time-T \rightarrow warningLightInput-T	input to pitch attitude sensor warning light
$\tilde{W}_\psi()$	time-T \rightarrow warningLightInput-T	input to heading sensor warning light

Table C.7: Constants

Symbol and Value	Type	Description
Generic constants		
$R = 287.05$	gasConstant-T	specific gas constant of the air
$T_0 = 288.15$	temperature-T	air temperature at sea level
$\lambda = -0.0065$	temperatureGradient-T	temperature gradient in troposphere
$g_0 = 9.80665$	acceleration-T	gravity acceleration at sea level
$p_0 = 101325$	pressure-T	air pressure at sea level
DHC-2 dynamics constants		
$C_{V_a} = 1$	TCWeight-T	weight coefficient for V_a term in trim-condition cost function
$C_{X_0} = -0.03554$	stabilityDerivative-T	stability derivative of force along X_B -axis
$C_{X_\alpha} = 0.00292$	stabilityDerivative-T	stability derivative of force along X_B -axis
$C_{X_q} = -0.6748$	stabilityDerivative-T	stability derivative of force along X_B -axis
$C_{X_{\alpha\delta f}} = 1.106$	controlDerivative-T	control derivative of force along X_B -axis
$C_{X_{\alpha dpt2}} = 0.1453$	controlDerivative-T	control derivative of force along Y_B -axis
$C_{X_{\alpha^2}} = 5.459$	stabilityDerivative-T	stability derivative of force along X_B -axis
$C_{X_{\alpha^3}} = -5.162$	stabilityDerivative-T	stability derivative of force along X_B -axis
$C_{X_{\delta f}} = -0.09447$	controlDerivative-T	control derivative of force along X_B -axis
$C_{X_{\delta r}} = 0.03412$	controlDerivative-T	control derivative of force along X_B -axis
$C_{X_{dpt}} = 0.1161$	controlDerivative-T	control derivative of force along X_B -axis
$C_{Y_0} = -0.002226$	stabilityDerivative-T	stability derivative of force along Y_B -axis
$C_{Y_\beta} = -0.7678$	stabilityDerivative-T	stability derivative of force along Y_B -axis
$C_{Y_p} = -0.124$	stabilityDerivative-T	stability derivative of force along Y_B -axis
$C_{Y_r} = 0.3666$	stabilityDerivative-T	stability derivative of force along Y_B -axis
$C_{Y_{\alpha\delta r}} = 0.5238$	controlDerivative-T	control derivative of force along Y_B -axis
$C_{Y_{\delta a}} = -0.02956$	controlDerivative-T	control derivative of force along Y_B -axis
$C_{Y_{\delta r}} = 0.1158$	controlDerivative-T	control derivative of force along Y_B -axis
$C_{Y_{\dot{\beta}}} = -0.16$	stabilityDerivative-T	stability derivative of force along Y_B -axis
$C_{Z_0} = -0.05504$	stabilityDerivative-T	stability derivative of force along Z_B -axis

Table C.7: Constants (continued)

Symbol and Value	Type	Description
$C_{Z_\alpha} = -5.578$	stabilityDerivative-T	stability derivative of force along Z_B -axis
$C_{Z_q} = -2.998$	stabilityDerivative-T	stability derivative of force along Z_B -axis
$C_{Z_{\alpha\delta f}} = -1.261$	controlDerivative-T	control derivative of force along Z_B -axis
$C_{Z_{\alpha^3}} = 3.442$	stabilityDerivative-T	stability derivative of force along Z_B -axis
$C_{Z_{\beta^2\delta_e}} = -15.93$	controlDerivative-T	control derivative of force along Z_B -axis
$C_{Z_{\delta_e}} = -0.398$	controlDerivative-T	control derivative of force along Z_B -axis
$C_{Z_{\delta f}} = -1.377$	controlDerivative-T	control derivative of force along Z_B -axis
$C_{Z_{dpt}} = -0.1563$	controlDerivative-T	control derivative of force along Z_B -axis
$C_{\alpha\beta} = 2$	TCWeight-T	weight coefficient for α and β terms in trim-condition cost function
$C_{l_0} = 0.000591$	stabilityDerivative-T	stability derivative of moment about X_B -axis
$C_{l_\beta} = -0.0618$	stabilityDerivative-T	stability derivative of moment about X_B -axis
$C_{l_p} = -0.5045$	stabilityDerivative-T	stability derivative of moment about X_B -axis
$C_{l_r} = 0.1695$	stabilityDerivative-T	stability derivative of moment about X_B -axis
$C_{l_{\alpha\delta a}} = -0.08269$	controlDerivative-T	control derivative of moment about X_B -axis
$C_{l_{\alpha^2dpt}} = -0.01406$	controlDerivative-T	control derivative of moment about X_B -axis
$C_{l_{\delta a}} = -0.09917$	controlDerivative-T	control derivative of moment about X_B -axis
$C_{l_{\delta r}} = 0.006934$	controlDerivative-T	control derivative of moment about X_B -axis
$C_{m_0} = 0.09448$	stabilityDerivative-T	stability derivative of moment about Y_B -axis
$C_{m_\alpha^*} = -0.6028$	stabilityDerivative-T	nominal value of stability derivative of moment about Y_B -axis
$C_{m_\alpha^+} = -0.428$	stabilityDerivative-T	maximum value of stability derivative of moment about Y_B -axis
$C_{m_\alpha^-} = -1.8$	stabilityDerivative-T	minimum value of stability derivative of moment about Y_B -axis
$C_{m_q} = -15.56$	stabilityDerivative-T	stability derivative of moment about Y_B -axis
$C_{m_r} = -0.3118$	stabilityDerivative-T	stability derivative of moment about Y_B -axis
$C_{m_{\alpha^2}} = -2.14$	stabilityDerivative-T	stability derivative of moment about Y_B -axis
$C_{m_{\beta^2}} = 0.6921$	stabilityDerivative-T	stability derivative of moment about Y_B -axis
$C_{m_{\delta_e}} = -1.921$	controlDerivative-T	control derivative of moment about Y_B -axis
$C_{m_{\delta f}} = 0.4072$	controlDerivative-T	control derivative of moment about Y_B -axis
$C_{m_{dpt}} = -0.07895$	controlDerivative-T	control derivative of moment about Y_B -axis

Table C.7: Constants (continued)

Symbol and Value	Type	Description
$C_{n_0} = -0.003117$	stabilityDerivative-T	stability derivative of moment about Z_B -axis
$C_{n_\beta} = 0.006719$	stabilityDerivative-T	stability derivative of moment about Z_B -axis
$C_{n_p} = -0.1585$	stabilityDerivative-T	stability derivative of moment about Z_B -axis
$C_{n_q} = 0.1595$	stabilityDerivative-T	stability derivative of moment about Z_B -axis
$C_{n_r} = -0.1112$	stabilityDerivative-T	stability derivative of moment about Z_B -axis
$C_{n_{\beta^3}} = 0.1373$	stabilityDerivative-T	stability derivative of moment about Z_B -axis
$C_{n_{\delta a}} = -0.003872$	controlDerivative-T	control derivative of moment about Z_B -axis
$C_{n_{\delta r}} = -0.08265$	controlDerivative-T	control derivative of moment about Z_B -axis
$C_{n_{dpt^3}} = -0.003026$	controlDerivative-T	control derivative of moment about Z_B -axis
$C_{pqr} = 5$	TCWeight-T	weight coefficient for p, q, r terms in trim-condition cost function
$\Delta C_{X_0}\% = 8\%$	percentage-T	uncertainty of stability derivative of force along X_B -axis
$\Delta C_{X_\alpha}\% = 8\%$	percentage-T	uncertainty of stability derivative of force along X_B -axis
$\Delta C_{X_q}\% = 16\%$	percentage-T	uncertainty of stability derivative of force along X_B -axis
$\Delta C_{X_{\alpha\delta f}}\% = 8\%$	percentage-T	uncertainty of control derivative of force along X_B -axis
$\Delta C_{X_{\alpha^2}}\% = 16\%$	percentage-T	uncertainty of stability derivative of force along X_B -axis
$\Delta C_{X_{\alpha^3}}\% = 16\%$	percentage-T	uncertainty of stability derivative of force along X_B -axis
$\Delta C_{X_{\delta f}}\% = 8\%$	percentage-T	uncertainty of control derivative of force along X_B -axis
$\Delta C_{X_{\delta r}}\% = 16\%$	percentage-T	uncertainty of control derivative of force along X_B -axis
$\Delta C_{Y_0}\% = 4\%$	percentage-T	uncertainty of stability derivative of force along Y_B -axis
$\Delta C_{Y_\beta}\% = 8\%$	percentage-T	uncertainty of stability derivative of force along Y_B -axis
$\Delta C_{Y_p}\% = 8\%$	percentage-T	uncertainty of stability derivative of force along Y_B -axis
$\Delta C_{Y_r}\% = 8\%$	percentage-T	uncertainty of stability derivative of force along Y_B -axis
$\Delta C_{Y_{\alpha\delta r}}\% = 16\%$	percentage-T	uncertainty of control derivative of force along Y_B -axis
$\Delta C_{Y_{\delta a}}\% = 8\%$	percentage-T	uncertainty of control derivative of force along Y_B -axis
$\Delta C_{Y_{\delta r}}\% = 8\%$	percentage-T	uncertainty of control derivative of force along Y_B -axis
$\Delta C_{Y_{\dot{\beta}}}\% = 16\%$	percentage-T	uncertainty of stability derivative of force along Y_B -axis
$\Delta C_{Z_0}\% = 8\%$	percentage-T	uncertainty of stability derivative of force along Z_B -axis
$\Delta C_{Z_\alpha}\% = 4\%$	percentage-T	uncertainty of stability derivative of force along Z_B -axis

Table C.7: Constants (continued)

Symbol and Value	Type	Description
$\Delta C_{Z_q}\% = 25\%$	percentage-T	uncertainty of stability derivative of force along Z_B -axis
$\Delta C_{Z_{\alpha\delta f}}\% = 8\%$	percentage-T	uncertainty of control derivative of force along Z_B -axis
$\Delta C_{Z_{\alpha 3}}\% = 16\%$	percentage-T	uncertainty of stability derivative of force along Z_B -axis
$\Delta C_{Z_{\beta^2\delta e}}\% = 25\%$	percentage-T	uncertainty of control derivative of force along Z_B -axis
$\Delta C_{Z_{\delta e}}\% = 4\%$	percentage-T	uncertainty of control derivative of force along Z_B -axis
$\Delta C_{Z_{\delta f}}\% = 8\%$	percentage-T	uncertainty of control derivative of force along Z_B -axis
$\Delta C_{l_0}\% = 8\%$	percentage-T	uncertainty of stability derivative of moment about X_B -axis
$\Delta C_{l_\beta}\% = 16\%$	percentage-T	uncertainty of stability derivative of moment about X_B -axis
$\Delta C_{l_p}\% = 16\%$	percentage-T	uncertainty of stability derivative of moment about X_B -axis
$\Delta C_{l_r}\% = 16\%$	percentage-T	uncertainty of stability derivative of moment about X_B -axis
$\Delta C_{l_{\alpha\delta a}}\% = 16\%$	percentage-T	uncertainty of control derivative of moment about X_B -axis
$\Delta C_{l_{\delta a}}\% = 8\%$	percentage-T	uncertainty of control derivative of moment about X_B -axis
$\Delta C_{l_{\delta r}}\% = 8\%$	percentage-T	uncertainty of control derivative of moment about X_B -axis
$\Delta C_{m_0}\% = 8\%$	percentage-T	uncertainty of stability derivative of moment about Y_B -axis
$\Delta C_{m_q}\% = 25\%$	percentage-T	uncertainty of stability derivative of moment about Y_B -axis
$\Delta C_{m_r}\% = 25\%$	percentage-T	uncertainty of stability derivative of moment about Y_B -axis
$\Delta C_{m_{\alpha^2}}\% = 16\%$	percentage-T	uncertainty of stability derivative of moment about Y_B -axis
$\Delta C_{m_{\beta^2}}\% = 16\%$	percentage-T	uncertainty of stability derivative of moment about Y_B -axis
$\Delta C_{m_{\delta e}}\% = 8\%$	percentage-T	uncertainty of control derivative of moment about Y_B -axis
$\Delta C_{m_{\delta f}}\% = 8\%$	percentage-T	uncertainty of control derivative of moment about Y_B -axis
$\Delta C_{n_0}\% = 8\%$	percentage-T	uncertainty of stability derivative of moment about Z_B -axis
$\Delta C_{n_\beta}\% = 8\%$	percentage-T	uncertainty of stability derivative of moment about Z_B -axis
$\Delta C_{n_p}\% = 16\%$	percentage-T	uncertainty of stability derivative of moment about Z_B -axis
$\Delta C_{n_q}\% = 25\%$	percentage-T	uncertainty of stability derivative of moment about Z_B -axis
$\Delta C_{n_r}\% = 16\%$	percentage-T	uncertainty of stability derivative of moment about Z_B -axis
$\Delta C_{n_{\beta 3}}\% = 25\%$	percentage-T	uncertainty of stability derivative of moment about Z_B -axis
$\Delta C_{n_{\delta a}}\% = 16\%$	percentage-T	uncertainty of control derivative of moment about Z_B -axis

Table C.7: Constants (continued)

Symbol and Value	Type	Description
$\Delta C_{n_{\delta r}}\% = 16\%$	percentage-T	uncertainty of control derivative of moment about Z_B -axis
$I_x = 5368.39$	momentOfInertia-T	moment of inertia about X_B -axis
$I_y = 6928.93$	momentOfInertia-T	moment of inertia about Y_B -axis
$I_z = 11158.75$	momentOfInertia-T	moment of inertia about Z_B -axis
$J_{xy} = 0$	productOfInertia-T	product of inertia in X_BY_B -plane
$J_{xz} = 117.64$	productOfInertia-T	product of inertia in X_BZ_B -plane
$J_{yz} = 0$	productOfInertia-T	product of inertia in Y_BZ_B -plane
$H_{at} = 2000 \cdot foot2SI$	altitude-T	minimum altitude value for atmospheric turbulence model to be valid
$\Delta I_x\% = 8\%$	percentage-T	uncertainty of moment of inertia about X_B -axis
$\Delta I_y\% = 8\%$	percentage-T	uncertainty of moment of inertia about Y_B -axis
$\Delta I_z\% = 8\%$	percentage-T	uncertainty of moment of inertia about Z_B -axis
$\Delta J_{xy}\% = 0\%$	percentage-T	uncertainty of product of inertia in X_BY_B -plane
$\Delta J_{xz}\% = 8\%$	percentage-T	uncertainty of product of inertia in X_BZ_B -plane
$\Delta J_{yz}\% = 0\%$	percentage-T	uncertainty of product of inertia in Y_BZ_B -plane
$S = 23.23$	area-T	wing area
$\bar{c} = 1.5875$	length-T	mean aerodynamic chord
$b = 14.63$	length-T	wing span
$m^- = 14970$	force-T	empty weight
$m^+ = 22800$	force-T	max-take-off weight
$r_x = 0.6, r_y = 0, r_z = 0$	length-T	component of IMU position along X_B, Y_B, Z_B axes
FCS hardware constants		
$ADC_{V-} = -10$	ADCinput-T	minimum value of ADC input
$ADC_{V+} = 10$	ADCinput-T	maximum value of ADC input
$ADC_{nb} = 16$	integer	ADC number of bits
$BIAS_A^- = -0.019$	accelerometerBias-T	minimum value of accelerometer bias
$BIAS_A^+ = +0.019$	accelerometerBias-T	maximum value of accelerometer bias
$BIAS_\alpha^- = -0.04$	AOTSensorBias-T	minimum value of angle of attack sensor bias

Table C.7: Constants (continued)

Symbol and Value	Type	Description
$BIAS_{\alpha}^{+} = +0.04$	AOTSensorBias-T	maximum value of angle of attack sensor bias
$BIAS_{\phi/\theta}^{-} = -0.01$	attitudeSensorBias-T	minimum value of attitude sensor bias
$BIAS_{\phi/\theta}^{+} = 0.01$	attitudeSensorBias-T	maximum value of attitude sensor bias
$BIAS_{\psi}^{-} = -0.04$	headingSensorBias-T	minimum value of heading sensor bias
$BIAS_{\psi}^{+} = 0.04$	headingSensorBias-T	maximum value of heading sensor bias
$BIAS_g^{-} = -0.05$	gyroBias-T	minimum value of rate gyro output bias
$BIAS_g^{+} = +0.05$	gyroBias-T	maximum value of rate gyro output bias
$BIAS_{p_s}^{-} = 0.95$	psSensorBias-T	minimum value of static pressure sensor bias
$BIAS_{p_s}^{+} = 1.05$	psSensorBias-T	maximum value of static pressure sensor bias
$BIAS_{q_{dyn}}^{-} = 0.95$	qdynSensorBias-T	minimum value of dynamic pressure sensor bias
$BIAS_{q_{dyn}}^{+} = 1.05$	qdynSensorBias-T	maximum value of dynamic pressure sensor bias
$BIAS_T^{-} = -0.05$	TSensorBias-T	minimum value of rate temperature sensor bias
$BIAS_T^{+} = +0.05$	TSensorBias-T	maximum value of rate temperature sensor bias
$BIAS_{\phi_r} = 0$	electricPotentialDifference-T	bias of attitude control output
$BIAS_{\psi_r} = 0$	electricPotentialDifference-T	bias of heading control output
$BIAS_{\theta_r} = 0$	electricPotentialDifference-T	bias of pitch control output
$DAC_{V-} = -10$	DACoutput-T	minimum value of DAC output
$DAC_{V+} = 10$	DACoutput-T	maximum value of DAC output
$DAC_{nb} = 16$	integer	DAC number of bits
$IR_A^{-} = -5 \cdot g2SI$	acceleration-T	minimum value of accelerometer input
$IR_A^{+} = 15 \cdot g2SI$	acceleration-T	maximum value of accelerometer input
$IR_{\alpha}^{-} = -30 \cdot deg2SI$	angle-T	minimum value of angle of attack sensor input
$IR_{\alpha}^{+} = 30 \cdot deg2SI$	angle-T	maximum value of angle of attack sensor input
$IR_{\phi/\theta}^{-} = -50 \cdot deg2SI$	angle-T	minimum value of attitude sensor input
$IR_{\phi/\theta}^{+} = 50 \cdot deg2SI$	angle-T	maximum value of attitude sensor input
$IR_{\psi}^{-} = -50 \cdot deg2SI$	angle-T	minimum value of heading sensor input
$IR_{\psi}^{+} = 50 \cdot deg2SI$	angle-T	maximum value of heading sensor input

Table C.7: Constants (continued)

Symbol and Value	Type	Description
$IR_g^- = -100 \cdot deg2SI$	angularVelocity-T	minimum value of rate gyros input
$IR_g^+ = 100 \cdot deg2SI$	angularVelocity-T	maximum value of rate gyros input
$IR_{ps}^- = 0$	pressure-T	minimum value of static pressure sensor input
$IR_{ps}^+ = 15 \cdot psi2SI$	pressure-T	maximum value of static pressure sensor input
$IR_{qdyn}^- = 0$	pressure-T	minimum value of dynamic pressure sensor input
$IR_{qdyn}^+ = 5 \cdot psi2SI$	pressure-T	maximum value of dynamic pressure sensor input
$IR_T^- = -40$	temperature-T	minimum value of temperature sensor input
$IR_T^+ = 80$	temperature-T	maximum value of temperature sensor input
$J_{tc}^+ = 10^{-4}$	float	maximum allowed value of trim-condition cost function
$Npsd_A = 0.000049$	sensorOutputPSD-T	noise power spectral density of accelerometer output
$Npsd_T = 0.000001$	sensorOutputPSD-T	noise power spectral density of temperature sensor output
$Npsd_\alpha = 0.000001$	sensorOutputPSD-T	noise power spectral density of angle of attack sensor output
$Npsd_{\phi/\theta} = 0.000001$	sensorOutputPSD-T	noise power spectral density of attitude sensor output
$Npsd_\psi = 0.000001$	sensorOutputPSD-T	noise power spectral density of heading sensor output
$Npsd_g = 0.0000000625$	sensorOutputPSD-T	noise power spectral density of rate gyro outputs
$Npsd_{ps} = 0.000001$	sensorOutputPSD-T	noise power spectral density of static pressure sensor output
$Npsd_{qdyn} = 0.000001$	sensorOutputPSD-T	noise power spectral density of dynamic pressure sensor output
$Npsd_{u_{wt}} = 1$	randomTurbulencePSD-T	power spectral density of noise driving Dryden turbulence model
$Npsd_{v_{wt}} = 1$	randomTurbulencePSD-T	power spectral density of noise driving Dryden turbulence model
$Npsd_{w_{wt}} = 1$	randomTurbulencePSD-T	power spectral density of noise driving Dryden turbulence model
$OR_A^- = -7.5$	electricPotentialDifference-T	minimum value of accelerometer output
$OR_A^+ = 7.5$	electricPotentialDifference-T	maximum value of accelerometer output
$OR_\alpha^- = -5$	electricPotentialDifference-T	minimum value of angle of attack sensor output
$OR_\alpha^+ = 5$	electricPotentialDifference-T	maximum value of angle of attack sensor output
$OR_{\phi/\theta}^- = -5$	electricPotentialDifference-T	minimum value of attitude sensor output
$OR_{\phi/\theta}^+ = 5$	electricPotentialDifference-T	maximum value of attitude sensor output
$OR_\psi^- = -5$	electricPotentialDifference-T	minimum value of heading sensor output
$OR_\psi^+ = 5$	electricPotentialDifference-T	maximum value of heading sensor output

Table C.7: Constants (continued)

Symbol and Value	Type	Description
$OR_g^- = -2.5$	electricPotentialDifference-T	minimum value of rate gyro outputs
$OR_g^+ = 2.5$	electricPotentialDifference-T	maximum value of rate gyro outputs
$OR_{p_s}^- = 0$	electricPotentialDifference-T	minimum value of static pressure sensor output
$OR_{p_s}^+ = 8$	electricPotentialDifference-T	maximum value of static pressure sensor output
$OR_{q_{dyn}}^- = 0$	electricPotentialDifference-T	minimum value of dynamic pressure sensor output
$OR_{q_{dyn}}^+ = 8$	electricPotentialDifference-T	maximum value of dynamic pressure sensor output
$OR_T^- = -10$	electricPotentialDifference-T	minimum value of temperature sensor output
$OR_T^+ = 10$	electricPotentialDifference-T	maximum value of temperature sensor output
$S_A^- = 1.485/g2SI$	acceleromterGain-T	minimum value of accelerometer gain
$S_A^+ = 1.515/g2SI$	acceleromterGain-T	maximum value of accelerometer gain
$S_T^- = 0.16$	TSensorGain-T	minimum value of temperature sensor gain
$S_T^+ = 0.17$	TSensorGain-T	maximum value of temperature sensor gain
$S_\alpha^- = 0.165/deg2SI$	AOTsensorGain-T	minimum value of angle of attack sensor gain
$S_\alpha^+ = 0.168/deg2SI$	AOTsensorGain-T	maximum value of angle of attack sensor gain
$S_{\phi/\theta}^- = 0.101/deg2SI$	attitudeSensorGain-T	minimum value of attitude sensor gain
$S_{\phi/\theta}^+ = 0.099/deg2SI$	attitudeSensorGain-T	maximum value of attitude sensor gain
$S_\psi^- = 0.101/deg2SI$	attitudeSensorGain-T	minimum value of heading sensor gain
$S_\psi^+ = 0.099/deg2SI$	attitudeSensorGain-T	maximum value of heading sensor gain
$S_g^- = 0.02475/deg2SI$	gyroGain-T	minimum value of rate gyro gains
$S_g^+ = 0.02525/deg2SI$	gyroGain-T	maximum value of rate gyro gains
$S_{p_s}^- = 0.330/psi2SI$	pressureSensorGain-T	minimum value of static pressure sensor gain
$S_{p_s}^+ = 0.337/psi2SI$	pressureSensorGain-T	maximum value of static pressure sensor gain
$S_{q_{dyn}}^- = 0.99/psi2SI$	pressureSensorGain-T	minimum value of dynamic pressure sensor gain
$S_{q_{dyn}}^+ = 1.01/psi2SI$	pressureSensorGain-T	maximum value of dynamic pressure sensor gain
$S\phi_r = 3/deg2SI$	CPGain-T	gain of reference bank control
$S\psi_r = 18/deg2SI$	CPGain-T	gain of reference heading control
$S\theta_r = 6.5/deg2SI$	CPGain-T	gain of reference pitching control

Table C.7: Constants (continued)

Symbol and Value	Type	Description
$T_s = 1/50$	time-T	sampling time
$V_{th} = 0$	CPOutput-T	threshold value of control switches
$\phi_r^- = -30 \cdot \text{deg2SI}$	angle-T	minimum value of roll angle reference
$\phi_r^+ = 30 \cdot \text{deg2SI}$	angle-T	maximum value of roll angle reference
$\psi_r^- = 0 \cdot \text{deg2SI}$	angle-T	minimum value of heading reference
$\psi_r^+ = 360 \cdot \text{deg2SI}$	angle-T	maximum value of heading reference
$\theta_r^- = -8 \cdot \text{deg2SI}$	angle-T	minimum value of pitch angle reference
$\theta_r^+ = 18 \cdot \text{deg2SI}$	angle-T	maximum value of pitch angle reference
$n^+ = 2300 \cdot \text{RPM2SI}$	engineSpeed-T	maximum value of engine speed
$p_z^+ = 26 \cdot \text{inHg2SI}$	pressure-T	maximum value of engine manifold pressure
$P^+ = 450 \cdot \text{hp2SI}$	power-T	maximum value of engine power
$\mathbf{A_r} = \begin{bmatrix} -9.2131 & 4.8550 & 14.0889 \\ 0 & 0 & 1.0000 \\ 0.6720 & -809.2957 & -50.3478 \end{bmatrix}$		state matrix for small perturbation model of rudder actuator
$\mathbf{B_r} = \begin{bmatrix} 24.5709 & 0.0042 \\ 0 & 0 \\ 7.5436 & -0.7416 \end{bmatrix}$		control matrix for small perturbation model of rudder actuator
$\mathbf{C_r} = \begin{bmatrix} 0 & 57.2958 & 0 \end{bmatrix}, \mathbf{D_r} = \begin{bmatrix} 0 & 0 \end{bmatrix}$		output matrices for small perturbation model of rudder actuator
$\mathbf{A_a} = \begin{bmatrix} -10.5970 & -3.2326 & -9.2515 \\ 0 & 0 & 1.0000 \\ 1.0877 & -684.8274 & -19.6229 \end{bmatrix}$		state matrix for small perturbation model of aileron actuator
$\mathbf{B_a} = \begin{bmatrix} 27.4630 & -0.0054 \\ 0 & 0 \\ 3.0187 & -1.1144 \end{bmatrix}$		control matrix for small perturbation model of aileron actuator
$\mathbf{C_a} = \begin{bmatrix} 0 & 83.8753 & 0 \end{bmatrix}, \mathbf{D_a} = \begin{bmatrix} 0 & 0 \end{bmatrix}$		output matrices for small perturbation model of aileron actuator

Table C.7: Constants (continued)

Symbol and Value	Type	Description
$\mathbf{A_e} = \begin{bmatrix} -10.9510 & 8.8068 & 25.1856 \\ 0 & 0 & 1.0000 \\ 7.3446 & -809.7962 & -20.1972 \end{bmatrix}$ $\mathbf{B_e} = \begin{bmatrix} 25.1568 & 0.0005 \\ 0 & 0 \\ 5.8694 & -0.0516 \end{bmatrix}$ $\mathbf{C_e} = \begin{bmatrix} 0 & 67.1621 & 0 \end{bmatrix}, \mathbf{D_e} = \begin{bmatrix} 0 & 0 \end{bmatrix}$ $C_{e1} = 0.08696, C_{e2} = 191.18, C_{e3} = 0.7355,$ $C_{e4} = -326.5, C_{e5} = 0.00412, C_{e6} = 7.4,$ $C_{e7} = 2010, C_{e8} = 408.0, C_{e9} = -0.0965$		state matrix for small perturbation model of elevator actuator control matrix for small perturbation model of elevator actuator output matrices for small perturbation model of elevator actuator engine model coefficients
FCS software constants		
$K_r = -4$ $K_{a-i} = 0.25$ $K_{s-i} = 0.5$ $\Delta\phi_{AL} = 0.03491$ $\Delta\phi_{PAH} = 0$ $\Delta\phi_V^+ = 3.75 \cdot \text{deg2SI}$ $\Delta\phi_V^- = -3.75 \cdot \text{deg2SI}$ $\Delta\phi_r^+ = 30 \cdot \text{deg2SI}$ $\Delta\phi_r^- = -30 \cdot \text{deg2SI}$ $\Delta\theta_V^+ = 2 \cdot \text{deg2SI}$ $\Delta\theta_V^- = -2 \cdot \text{deg2SI}$ $\Delta\theta_r^+ = 18 \cdot \text{deg2SI}$ $\Delta\theta_r^- = -8 \cdot \text{deg2SI}$ $dar = -0.165$ $l_a^- = -1 \cdot \text{deg2SI}$ $l_a^+ = 1 \cdot \text{deg2SI}$	float frequency-T frequency-T angle-T angle-T angle-T angle-T angle-T angle-T angle-T angle-T angle-T angle-T float angle-T angle-T	constant of asymmetric autopilot integration gain of asymmetric autopilot integration gain of symmetric autopilot correction term in ALH autopilot correction term in PAH autopilot max allowed value for $\Delta\phi$ min allowed value for $\Delta\phi$ $\Delta\phi_r$ minimum value $\Delta\phi_r$ minimum value max allowed value for $\Delta\theta$ min allowed value for $\Delta\theta$ $\Delta\theta_r$ minimum value $\Delta\theta_r$ minimum value turn-coordination loop constant of asymmetric autopilot inferior limit of asymmetric autopilot integrator superior limit of asymmetric autopilot integrator

Table C.7: Constants (continued)

Symbol and Value	Type	Description
$l_{s1}^- = -10 \cdot deg2SI$	angle-T	inferior limit of symmetric autopilot integrator
$l_{s1}^+ = 10 \cdot deg2SI$	angle-T	superior limit of symmetric autopilot integrator
$l_{s2}^- = -20 \cdot deg2SI$	angle-T	inferior limit of ALH autopilot
$l_{s2}^+ = 20 \cdot deg2SI$	angle-T	superior limit of ALH autopilot
Conversion factors		
$psi2SI = 6894.757$	float	conversion factor from <i>psi</i> to <i>Pa</i>
$foot2SI = 0.3048$	float	conversion factor from <i>foot</i> to <i>m</i>
$deg2SI = \pi/180$	float	conversion factor from <i>degree</i> to <i>rad</i>
$g2SI = 9.80665$	float	conversion factor from <i>g</i> to $m \cdot s^{-2}$
$RPM2SI = 0.1047198$	float	conversion factor from <i>RPM</i> to $rad \cdot s^{-1}$
$inHg2SI = 3386.389$	float	conversion factor from <i>inHg</i> to <i>Pa</i>
$hp2SI = 745.6999$	float	conversion factor from <i>hp</i> to $J \cdot s^{-1}$

Table C.8: Quantified variables

Symbol	Type	Description
$C_{X_0}()$	time-T \rightarrow stabilityDerivative-T	actual value of stability derivative of force along X_B -axis
$C_{X_\alpha}()$	time-T \rightarrow stabilityDerivative-T	actual value of stability derivative of force along X_B -axis
$C_{X_q}()$	time-T \rightarrow stabilityDerivative-T	actual value of stability derivative of force along X_B -axis
$C_{X_{\alpha\delta f}}()$	time-T \rightarrow controlDerivative-T	actual value of control derivative of force along X_B -axis
$C_{X_{\alpha^2}}()$	time-T \rightarrow stabilityDerivative-T	actual value of stability derivative of force along X_B -axis
$C_{X_{\alpha^3}}()$	time-T \rightarrow stabilityDerivative-T	actual value of stability derivative of force along X_B -axis
$C_{X_{\delta f}}()$	time-T \rightarrow controlDerivative-T	actual value of control derivative of force along X_B -axis
$C_{X_{\delta r}}()$	time-T \rightarrow controlDerivative-T	actual value of control derivative of force along X_B -axis
$C_{Y_0}()$	time-T \rightarrow stabilityDerivative-T	actual value of stability derivative of force along Y_B -axis
$C_{Y_\beta}()$	time-T \rightarrow stabilityDerivative-T	actual value of stability derivative of force along Y_B -axis
$C_{Y_p}()$	time-T \rightarrow stabilityDerivative-T	actual value of stability derivative of force along Y_B -axis
$C_{Y_r}()$	time-T \rightarrow stabilityDerivative-T	actual value of stability derivative of force along Y_B -axis
$C_{Y_{\alpha\delta r}}()$	time-T \rightarrow controlDerivative-T	actual value of control derivative of force along Y_B -axis
$C_{Y_{\delta a}}()$	time-T \rightarrow controlDerivative-T	actual value of control derivative of force along Y_B -axis
$C_{Y_{\delta r}}()$	time-T \rightarrow controlDerivative-T	actual value of control derivative of force along Y_B -axis
$C_{Y_{\dot{\beta}}}()$	time-T \rightarrow stabilityDerivative-T	actual value of stability derivative of force along Y_B -axis
$C_{Z_0}()$	time-T \rightarrow stabilityDerivative-T	actual value of stability derivative of force along Z_B -axis
$C_{Z_\alpha}()$	time-T \rightarrow stabilityDerivative-T	actual value of stability derivative of force along Z_B -axis
$C_{Z_q}()$	time-T \rightarrow stabilityDerivative-T	actual value of stability derivative of force along Z_B -axis
$C_{Z_{\alpha\delta f}}()$	time-T \rightarrow controlDerivative-T	actual value of control derivative of force along Z_B -axis
$C_{Z_{\alpha^3}}()$	time-T \rightarrow stabilityDerivative-T	actual value of stability derivative of force along Z_B -axis
$C_{Z_{\beta^2\delta e}}()$	time-T \rightarrow controlDerivative-T	actual value of control derivative of force along Z_B -axis
$C_{Z_{\delta e}}()$	time-T \rightarrow controlDerivative-T	actual value of control derivative of force along Z_B -axis
$C_{Z_{\delta f}}()$	time-T \rightarrow controlDerivative-T	actual value of control derivative of force along Z_B -axis
$C_{l_0}()$	time-T \rightarrow stabilityDerivative-T	actual value of stability derivative of moment about X_B -axis
$C_{l_\beta}()$	time-T \rightarrow stabilityDerivative-T	actual value of stability derivative of moment about X_B -axis
$C_{l_p}()$	time-T \rightarrow stabilityDerivative-T	actual value of stability derivative of moment about X_B -axis
$C_{l_r}()$	time-T \rightarrow stabilityDerivative-T	actual value of stability derivative of moment about X_B -axis

Table C.8: Quantified variables (continued)

Symbol	Type	Description
$C_{l_{\alpha\delta a}}()$	time-T \rightarrow controlDerivative-T	actual value of control derivative of moment about X_B -axis
$C_{l_{\delta a}}()$	time-T \rightarrow controlDerivative-T	actual value of control derivative of moment about X_B -axis
$C_{l_{\delta r}}()$	time-T \rightarrow controlDerivative-T	actual value of control derivative of moment about X_B -axis
$C_{m_0}()$	time-T \rightarrow stabilityDerivative-T	actual value of stability derivative of moment about Y_B -axis
$C_{m_{\alpha}}()$	time-T \rightarrow stabilityDerivative-T	actual value of stability derivative of moment about Y_B -axis
$C_{m_q}()$	time-T \rightarrow stabilityDerivative-T	actual value of stability derivative of moment about Y_B -axis
$C_{m_r}()$	time-T \rightarrow stabilityDerivative-T	actual value of stability derivative of moment about Y_B -axis
$C_{m_{\alpha^2}}()$	time-T \rightarrow stabilityDerivative-T	actual value of stability derivative of moment about Y_B -axis
$C_{m_{\beta^2}}()$	time-T \rightarrow stabilityDerivative-T	actual value of stability derivative of moment about Y_B -axis
$C_{m_{\delta e}}()$	time-T \rightarrow controlDerivative-T	actual value of control derivative of moment about Y_B -axis
$C_{m_{\delta f}}()$	time-T \rightarrow controlDerivative-T	actual value of control derivative of moment about Y_B -axis
$C_{n_0}()$	time-T \rightarrow stabilityDerivative-T	actual value of stability derivative of moment about Z_B -axis
$C_{n_{\beta}}()$	time-T \rightarrow stabilityDerivative-T	actual value of stability derivative of moment about Z_B -axis
$C_{n_p}()$	time-T \rightarrow stabilityDerivative-T	actual value of stability derivative of moment about Z_B -axis
$C_{n_q}()$	time-T \rightarrow stabilityDerivative-T	actual value of stability derivative of moment about Z_B -axis
$C_{n_r}()$	time-T \rightarrow stabilityDerivative-T	actual value of stability derivative of moment about Z_B -axis
$C_{n_{\beta^3}}()$	time-T \rightarrow stabilityDerivative-T	actual value of stability derivative of moment about Z_B -axis
$C_{n_{\delta a}}()$	time-T \rightarrow controlDerivative-T	actual value of control derivative of moment about Z_B -axis
$C_{n_{\delta r}}()$	time-T \rightarrow controlDerivative-T	actual value of control derivative of moment about Z_B -axis
$I_x()$	time-T \rightarrow momentOfInertia-T	actual value of moment of inertia along X_B -axis
$I_y()$	time-T \rightarrow momentOfInertia-T	actual value of moment of inertia along Y_B -axis
$I_z()$	time-T \rightarrow momentOfInertia-T	actual value of moment of inertia along Z_B -axis
$J_{xy}()$	time-T \rightarrow productOfInertia-T	actual value of product of inertia in X_BY_B -plane
$J_{xz}()$	time-T \rightarrow productOfInertia-T	actual value of product of inertia in X_BZ_B -plane
$J_{yz}()$	time-T \rightarrow productOfInertia-T	actual value of product of inertia in Y_BZ_B -plane
$P()$	time-T \rightarrow power-T	engine power
S_{A_x}	accelerometerGain-T	actual value of A_x accelerometer gain
S_{A_y}	accelerometerGain-T	actual value of A_y accelerometer gain

Table C.8: Quantified variables (continued)

Symbol	Type	Description
S_{A_z}	accelerometerGain-T	actual value of A_z accelerometer gain
S_T	TSensorGain-T	actual value of temperature sensor gain
S_α	AOTSensorGain-T	actual value of angle of attack sensor gain
S_ϕ	attitudeSensorGain-T	actual value of roll attitude sensor gain
S_ψ	attitudeSensorGain-T	actual value of heading sensor gain
S_θ	attitudeSensorGain-T	actual value of pitch attitude sensor gain
S_{p_s}	pressureSensorGain-T	actual value of static pressure sensor gain
S_p	gyroGain-T	actual value of roll rate gyro gain
$S_{q_{dyn}}$	pressureSensorGain-T	actual value of dynamic pressure sensor gain
S_q	gyroGain-T	actual value of pitch rate gyro gain
S_r	gyroGain-T	actual value of yaw rate gyro gain
$\mathbf{x}_a()$	array of float	state vector of small perturbation aileron model
$\mathbf{x}_e()$	array of float	state vector of small perturbation elevator model
$\mathbf{x}_r()$	array of float	state vector of small perturbation rudder model
$\nu_{A_x}()$	time-T \rightarrow accelerometerOutput-T	instance of A_x accelerometer output noise
$\nu_{A_y}()$	time-T \rightarrow accelerometerOutput-T	instance of A_y accelerometer output noise
$\nu_{A_z}()$	time-T \rightarrow accelerometerOutput-T	instance of A_z accelerometer output noise
$\nu_T()$	time-T \rightarrow TSensorOutput-T	instance of temperature sensor output noise
$\nu_\alpha()$	time-T \rightarrow AOTSensorOutput-T	instance of angle of attack sensor output noise
$\nu_\phi()$	time-T \rightarrow attitudeSensorOutput-T	instance of roll attitude sensor output noise
$\nu_\psi()$	time-T \rightarrow headingSensorOutput-T	instance of heading sensor output noise
$\nu_\theta()$	time-T \rightarrow attitudeSensorOutput-T	instance of pitch attitude sensor output noise
$\nu_{p_s}()$	time-T \rightarrow pressureSensorOutput-T	instance of static pressure sensor output noise
$\nu_p()$	time-T \rightarrow gyroOutput-T	instance of roll rate gyro output noise
$\nu_{q_{dyn}}()$	time-T \rightarrow pressureSensorOutput-T	instance of dynamic pressure sensor output noise
$\nu_q()$	time-T \rightarrow gyroOutput-T	instance of pitch rate gyro output noise
$\nu_r()$	time-T \rightarrow gyroOutput-T	instance of yaw rate gyro output noise
$\nu_{w_{wt}}()$	time-T \rightarrow float	instance of noise driving Dryden turbulence model

Table C.8: Quantified variables (continued)

Symbol	Type	Description
$\nu_{v_{wt}}()$	time-T \rightarrow float	instance of noise driving Dryden turbulence model
$\nu_{w_{wt}}()$	time-T \rightarrow float	instance of noise driving Dryden turbulence model
a, b	float	coefficients used within the trim-condition requirement
$bias_{A_x}$	accelerometerBias-T	actual value of A_x accelerometer bias
$bias_{A_y}$	accelerometerBias-T	actual value of A_y accelerometer bias
$bias_{A_z}$	accelerometerBias-T	actual value of A_z accelerometer bias
$bias_T$	TSensorBias-T	actual value of temperature sensor bias
$bias_\alpha$	AOTSensorBias-T	actual value of angle of attack sensor bias
$bias_\phi$	attitudeSensorBias-T	actual value of roll attitude sensor bias
$bias_\psi$	headingSensorBias-T	actual value of heading sensor bias
$bias_\theta$	attitudeSensorBias-T	actual value of pitch attitude sensor bias
$bias_{p_s}$	psSensorBias-T	actual value of static pressure sensor bias
$bias_p$	gyroBias-T	actual value of roll rate gyro bias
$bias_{q_{dyn}}$	qdynSensorBias-T	actual value of dynamic pressure sensor bias
$bias_q$	gyroBias-T	actual value of pitch rate gyro bias
$bias_r$	gyroBias-T	actual value of yaw rate gyro bias
$dpt()$	time-T \rightarrow float	non-dimensional pressure increase in propeller slipstream
k	integer	counter
$m()$	time-T \rightarrow force-T	actual airplane weight
t, t_1, t_2, t_3, t_4	time-T	continuous time instant

Table C.9: Auxiliary terms

Symbol and expression in terms of base quantities	Type	Description
Auxiliary terms		
$F_x() = X_{asd}() + X_{acd}() + X_p() + X_{gr}() + X_w()$	time-T \rightarrow force-T	total force along X_B -axis
$F_y() = Y_{asd}() + Y_{acd}() + Y_p() + Y_{gr}() + Y_w()$	time-T \rightarrow force-T	total force along Y_B -axis
$F_z() = Z_{asd}() + Z_{acd}() + Z_p() + Z_{gr}() + Z_w()$	time-T \rightarrow force-T	total force along Z_B -axis
$I_1() = I_y()I_z() - J_{yz}^2()$	time-T \rightarrow inertiaParam-T	inertia parameter for moment equations
$I_2() = J_{xy}()I_z() + J_{yz}()J_{xz}()$	time-T \rightarrow inertiaParam-T	inertia parameter for moment equations
$I_3() = J_{xy}()J_{yz}() + I_y()J_{xz}()$	time-T \rightarrow inertiaParam-T	inertia parameter for moment equations
$I_4() = I_x()I_z() - J_{xz}^2()$	time-T \rightarrow inertiaParam-T	inertia parameter for moment equations
$I_5() = I_x()J_{yz}() + J_{xy}()J_{xz}()$	time-T \rightarrow inertiaParam-T	inertia parameter for moment equations
$I_6() = I_x()I_y() - J_{xy}^2()$	time-T \rightarrow inertiaParam-T	inertia parameter for moment equations
$ I() = I_x()I_y()I_z() - 2J_{xy}()J_{xz}()J_{yz}() - I_x()J_{yz}^2() - I_y()J_{xz}^2() - I_z()J_{xy}^2()$	time-T \rightarrow inertiaParam-T	inertia parameter for moment equations
$J_{tc}((\dot{V}_a(0), \dot{\alpha}(0), \dot{\beta}(0), \dot{p}(0), \dot{q}(0), \dot{r}(0))) = C_{V_a}\dot{V}_a^2(0) + C_{\alpha\beta}(\dot{\alpha}^2(0) + \dot{\beta}^2(0)) + C_{pqr}(\dot{p}^2(0) + \dot{q}^2(0) + \dot{r}^2(0))$	(acceleration-T, angularVelocity-T, angularVelocity-T, angularAcceleration-T, angularAcceleration-T, angularAcceleration-T) \rightarrow float	cost function for trim-condition
$K_H(\hat{V}_a) = (-1 \cdot 10^{-4}V_a^2 + 0.015\hat{V}_a - 0.5975)\pi/180$	airspeed-T \rightarrow float	proportional gain of ALH autopilot
$K_\phi(\hat{V}_a) = 9.75 \cdot 10^{-4}\hat{V}_a^2 - 0.108\hat{V}_a + 2.335625$	airspeed-T \rightarrow float	proportional gain of asymmetric autopilot
$K_\psi(\hat{V}_a) = 0.05\hat{V}_a - 1.1$	airspeed-T \rightarrow float	proportional gain of HH/HS autopilot
$K_\theta(\hat{V}_a) = 1.375 \cdot 10^{-3}\hat{V}_a^2 + 0.1575\hat{V}_a - 4.8031$	airspeed-T \rightarrow float	proportional gain of PAH autopilot
$\bar{K}_\theta(\hat{V}_a) = 1.375 \cdot 10^{-3}\hat{V}_a^2 + 0.1575\hat{V}_a - 4.8031$	airspeed-T \rightarrow float	proportional gain of ALH and ALS autopilot
$K_d(\hat{V}_a) = -2.5 \cdot 10^{-3}\hat{V}_a + 0.2875$	airspeed-T \rightarrow float	proportional gain of ALH and ALS autopilot

Table C.9: Auxiliary terms (continued)

Symbol and expression in terms of base quantities	Type	Description
$K_q(\hat{V}_a) = -4.75 \cdot 10^{-4} \hat{V}_a^2 + 0.0540 \hat{V}_a - 1.593$	airspeed-T \rightarrow float	proportional gain of symmetric autopilot
$K_{\dot{H}}(\hat{V}_a) = (-3.875 \cdot 10^{-4} \hat{V}_a^2 + 0.04025 \hat{V}_a - 1.1041)\pi/180$	airspeed-T \rightarrow float	proportional gain of ALS autopilot
$K_{tc}(\hat{V}_a) = 5 \cdot 10^{-4} \hat{V}_a^2 - 0.03 \hat{V}_a + 0.9375$	airspeed-T \rightarrow float	proportional gain of PAH autopilot
$\bar{K}_{tc}(\hat{V}_a) = 0.03 \hat{V}_a + 0.25$	airspeed-T \rightarrow float	proportional gain of ALH and ALS autopilot
$L() = L_{a_{sd}}() + L_{a_{cd}}() + L_p()$	time-T \rightarrow moment-T	total rolling moment
$M() = M_{a_{sd}}() + M_{a_{cd}}() + M_p()$	time-T \rightarrow moment-T	total pitching moment
$N() = N_{a_{sd}}() + N_{a_{cd}}() + N_p()$	time-T \rightarrow moment-T	total yawing moment
$OFF_{ADC} = 2^{ADC_{nb}-1}$	integer	ADC internal representation of ADC zero input
$OFF_{DAC} = 2^{DAC_{nb}-1}$	integer	DAC internal representation of DAC zero output
$P_l() = I_1()/ I() $	time-T \rightarrow inertiaParam-T	inertia parameter for moment equations
$P_m() = I_2()/ I() $	time-T \rightarrow inertiaParam-T	inertia parameter for moment equations
$P_n() = I_3()/ I() $	time-T \rightarrow inertiaParam-T	inertia parameter for moment equations
$P_{pp}() = -(J_{xz}()I_2() - J_{xy}()I_3())/ I() $	time-T \rightarrow inertiaParam-T	inertia parameter for moment equations
$P_{pq}() = (J_{xz}()I_1() - J_{yz}()I_2() - (I_y() - I_x())I_3())/ I() $	time-T \rightarrow inertiaParam-T	inertia parameter for moment equations
$P_{pr}() = -(J_{xy}()I_1() + (I_x() - I_z())I_2() - J_{yz}()I_3())/ I() $	time-T \rightarrow inertiaParam-T	inertia parameter for moment equations
$P_{qq}() = (J_{yz}()I_1() - J_{xy}()I_3())/ I() $	time-T \rightarrow inertiaParam-T	inertia parameter for moment equations
$P_{qr}() = -((I_z() - I_y())I_1() - J_{xy}()I_2() + J_{xz}()I_3())/ I() $	time-T \rightarrow inertiaParam-T	inertia parameter for moment equations
$P_{rr}() = -(J_{yz}()I_1() - J_{xz}()I_2())/ I() $	time-T \rightarrow inertiaParam-T	inertia parameter for moment equations
$Q_l() = I_2()/ I() $	time-T \rightarrow inertiaParam-T	inertia parameter for moment equations
$Q_m() = I_4()/ I() $	time-T \rightarrow inertiaParam-T	inertia parameter for moment equations
$Q_n() = I_5()/ I() $	time-T \rightarrow inertiaParam-T	inertia parameter for moment equations
$Q_{pp}() = -(J_{xz}()I_4() - J_{xy}()I_5())/ I() $	time-T \rightarrow inertiaParam-T	inertia parameter for moment equations

Table C.9: Auxiliary terms (continued)

Symbol and expression in terms of base quantities	Type	Description
$Q_{pq}() = (J_{xz}()I_2() - J_{yz}I_4() - (I_y() - I_x())I_5())/ I() $	time-T \rightarrow inertiaParam-T	inertia parameter for moment equations
$Q_{pr}() = -(J_{xy}()I_2() + (I_x() - I_z())I_4() - J_{yz}()I_5())/ I() $	time-T \rightarrow inertiaParam-T	inertia parameter for moment equations
$Q_{qq}() = (J_{yz}()I_2() - J_{xy}()I_5())/ I() $	time-T \rightarrow inertiaParam-T	inertia parameter for moment equations
$Q_{qr}() = -((I_z() - I_y())I_2() - J_{xy}()I_4() + J_{xz}()I_5())/ I() $	time-T \rightarrow inertiaParam-T	inertia parameter for moment equations
$Q_{rr}() = -(J_{yz}()I_2() - J_{xz}()I_4())/ I() $	time-T \rightarrow inertiaParam-T	inertia parameter for moment equations
$R_l() = I_3()/ I() $	time-T \rightarrow inertiaParam-T	inertia parameter for moment equations
$R_m() = I_5()/ I() $	time-T \rightarrow inertiaParam-T	inertia parameter for moment equations
$R_n() = I_6()/ I() $	time-T \rightarrow inertiaParam-T	inertia parameter for moment equations
$R_{pp}() = -(J_{xz}()I_5() - J_{xy}()I_6())/ I() $	time-T \rightarrow inertiaParam-T	inertia parameter for moment equations
$R_{pq}() = (J_{xz}()I_3() - J_{yz}()I_5() - (I_y() - I_x())I_6())/ I() $	time-T \rightarrow inertiaParam-T	inertia parameter for moment equations
$R_{pr}() = -(J_{xy}()I_3() + (I_x() - I_z())I_5() - J_{yz}()I_6())/ I() $	time-T \rightarrow inertiaParam-T	inertia parameter for moment equations
$R_{qq}() = (J_{yz}()I_3() - J_{xy}()I_6())/ I() $	time-T \rightarrow inertiaParam-T	inertia parameter for moment equations
$R_{qr}() = -((I_z() - I_y())I_3() - J_{xy}()I_5() + J_{xz}()I_6())/ I() $	time-T \rightarrow inertiaParam-T	inertia parameter for moment equations
$R_{rr}() = -(J_{yz}()I_3() - J_{xz}()I_5())/ I() $	time-T \rightarrow inertiaParam-T	inertia parameter for moment equations
$S_{ADC} = \frac{2^{ADC_{nb}-1}}{DAC_{V+}-DAC_{V-}}$	float	ADC resolution
$S_{DAC} = \frac{2^{DAC_{nb}-1}}{DAC_{V+}-DAC_{V-}}$	float	DAC resolution
$h_{u_{wt}}() = \mathcal{F}^{-1} \left\{ \sigma_u \sqrt{\frac{2L_u}{V_a(t)}} \frac{1}{(1 + \frac{L_u}{V_a(t)} j\omega)} \right\}$	time-T \rightarrow turbulence-T	filter implementing Dryden turbulence model
$h_{v_{wt}}() = \mathcal{F}^{-1} \left\{ \sigma_v \sqrt{\frac{L_v}{V_a(t)}} \frac{1 + \sqrt{3} \frac{L_v}{V_a(t)} j\omega}{(1 + \frac{L_v}{V_a(t)} j\omega)^2} \right\}$	time-T \rightarrow turbulence-T	filter implementing Dryden turbulence model

Table C.9: Auxiliary terms (continued)

Symbol and expression in terms of base quantities	Type	Description
$h_{wt}() = \mathcal{F}^{-1} \left\{ \sigma_w \sqrt{\frac{L_w}{V_a(t)}} \frac{1 + \sqrt{3} \frac{L_w}{V_a(t)} j\omega}{\left(1 + \frac{L_w}{V_a(t)} j\omega\right)^2} \right\}$	time-T \rightarrow turbulence-T	filter implementing Dryden turbulence model
$\tilde{\mathbf{u}}_a() = \begin{bmatrix} \tilde{\delta}_a() & \frac{bp()}{2V_a()} \end{bmatrix}^T$	array of float	input vector to aileron model
$\tilde{\mathbf{u}}_e() = \begin{bmatrix} \tilde{\delta}_e() & \frac{\bar{c}q()}{V_a()} \end{bmatrix}^T$	array of float	input vector to elevator model
$\tilde{\mathbf{u}}_r() = \begin{bmatrix} \tilde{\delta}_r() & \frac{br()}{2V_a()} \end{bmatrix}^T$	array of float	input vector to rudder model
$\hat{\mathbf{u}}_a() = \begin{bmatrix} \hat{\delta}_a() & \frac{bp()}{2V_a()} \end{bmatrix}^T$	array of float	input vector to aileron model
$\hat{\mathbf{u}}_e() = \begin{bmatrix} \hat{\delta}_e() & \frac{\bar{c}q()}{V_a()} \end{bmatrix}^T$	array of float	input vector to elevator model
$\hat{\mathbf{u}}_r() = \begin{bmatrix} \hat{\delta}_r() & \frac{br()}{2V_a()} \end{bmatrix}^T$	array of float	input vector to rudder model
$drr(\hat{V}_a) = -7.5 \cdot 10^{-5} \hat{V}_a^2 - 0.0095 \hat{V}_a - 0.4606$	airspeed-T \rightarrow float	turn-coordination loop gain of asymmetric autopilot
$u_a() = V_a() \cos \alpha() \cos \beta()$	time-T \rightarrow velocity-T	airspeed component along X_B -axis
$u_e() = u_a() + u_w()$	time-T \rightarrow velocity-T	component along X_B -axis of velocity wrt earth frame
$v_a() = V_a \sin \beta()$	time-T \rightarrow velocity-T	airspeed component along Y_B -axis
$v_e() = v_a() + v_w()$	time-T \rightarrow velocity-T	component along Y_B -axis of velocity wrt earth frame
$w_a() = V_a() \sin \alpha() \cos \beta()$	time-T \rightarrow velocity-T	airspeed component along Z_B -axis
$w_e() = w_a() + w_w()$	time-T \rightarrow velocity-T	component along Z_B -axis of velocity wrt earth frame
$u_w() = u_{\bar{w}}() + u_{w_t}() + u_{w_g}()$	time-T \rightarrow velocity-T	wind velocity component along X_B -axis
$v_w() = v_{\bar{w}}() + v_{w_t}() + v_{w_g}()$	time-T \rightarrow velocity-T	wind velocity component along Y_B -axis
$w_w() = w_{\bar{w}}() + w_{w_t}() + w_{w_g}()$	time-T \rightarrow velocity-T	wind velocity component along Z_B -axis
$\widehat{\Delta H}() = \hat{H}() - H_0$	natural-T \rightarrow altitude-T	altitude variation with respect to trim value

Table C.9: Auxiliary terms (continued)

Symbol and expression in terms of base quantities	Type	Description
$\widehat{\Delta\delta_a}() = \hat{\delta}_a() - \delta_{a0}$	natural-T → aileronDeflection-T	aileron deflection variation with respect to trim value
$\widehat{\Delta\delta_e}() = \hat{\delta}_e() - \delta_{e0}$	natural-T → elevatorDeflection-T	elevator deflection variation with respect to trim value
$\widehat{\Delta\delta_r}() = \hat{\delta}_r() - \delta_{r0}$	natural-T → rudderDeflection-T	rudder deflection variation with respect to trim value
$\widehat{\Delta\phi}() = \hat{\phi}() - \phi_0$	natural-T → angle-T	bank angle variation with respect to trim value
$\widehat{\Delta\phi_r}() = \hat{\phi}_r() - \phi_{r0}$	natural-T → bankReference-T	bank reference variation with respect to trim value
$\widehat{\Delta\psi}() = \hat{\psi}() - \psi_0$	natural-T → angle-T	heading variation with respect to trim value
$\widehat{\Delta\psi_r}() = \hat{\psi}_r() - \psi_{r0}$	natural-T → headingReference-T	heading reference variation with respect to trim value
$\widehat{\Delta\theta}() = \hat{\theta}() - \theta_0$	natural-T → angle-T	pitch angle variation with respect to trim value
$\widehat{\Delta\theta_r}() = \hat{\theta}_r() - \theta_{r0}$	natural-T → pitchReference-T	pitch reference variation with respect to trim value
Auxiliary functions, operators, and predicates		
$constRef(f(), t_1, t_2) = \exists k \forall t \left(t_1 \leq t \leq t_2 \Rightarrow f(t) = k \right)$	not specified	predicate that evaluates true if $f()$ is constant throughout the time interval $[t_1, t_2]$
$engaged(SW(), t_1, t_2) = \exists T_\epsilon \left(0 < T_\epsilon < \infty \wedge \forall t \left(t_1 - T_\epsilon < t < t_1 \Rightarrow SW(t) = OFF \right) \wedge \forall t \left(t_1 \leq t < t_2 \Rightarrow SW(t) = ON \right) \right)$	(switch-T, time-T, time-T) → boolean	predicate that evaluates true only if the switch $SW()$ is engaged at $t = t_1$ and stays engaged throughout the time interval $[t_1, t_2]$
$turb(t_1, t_2) = \neg \forall t \left(t_1 \leq t \leq t_2 \Rightarrow u_{wt}(t) = 0 \wedge w_{wt}(t) = 0 \wedge u_{wg}(t) = 0 \wedge w_{wg}(t) = 0 \wedge w_{wg}(t) = 0 \wedge \right)$	(time-T, time-T) → boolean	predicate that evaluates true if random and discrete turbulence wind components are not zero

Table C.9: Auxiliary terms (continued)

Symbol and expression in terms of base quantities	Type	Description
$whileEngaged(SW(), t_1, t_2) = \forall t \left(t_1 \leq t < t_2 \Rightarrow SW(t) = ON \right)$	(switch-T, time-T, time-T) \rightarrow boolean	predicate that evaluates true only if the switch $SW()$ is engaged throughout the time interval $[t_1, t_2]$
$whiteNoise(\nu(), Npsd)$	not specified	predicate that evaluates true if $\nu()$ is white noise with Power Spectral Density $Npsd$
$p_1() \simeq p_2() \Leftrightarrow E\{a(t)a(t+\tau)\} = E\{b(t)b(t+\tau)\}$	not specified	equivalence operator between stochastic processes
$\mathbf{z}(f(k)) = f(k+1)$	not specified	one step forward operator
$\left[x \right]_a^b = \begin{cases} x & : a < x < b \\ a & : a \leq x \\ b & : x \leq b \end{cases}$	not specified	crop operator
$RMS(f(), t_1, t_2) = \sqrt{\frac{1}{t_2-t_1} \int_{t_1}^{t_2} f^2(t) dt}$	not specified	Root Mean Square value of a function over the interval $[t_1, t_2]$
$\lfloor x \rfloor = y \Leftrightarrow y \in \mathbf{N} \wedge y < x \wedge \forall n \left(n \in \mathbf{N} \wedge n > y \Rightarrow n > x \right)$	not specified	quantization to closest integer

Table C.10: Data-types

Type ID	Base type	Range	SI symbol	Description
ADCinput-T	electricPotentialDifference-T	$[ADC_{V-}, ADC_{V+}]$	<i>inherited</i>	input to A/D card
ADCoutput-T	integer	$[0, 2^{ADC_{nb}})$	<i>inherited</i>	output from A/D card
AOT-T	angle-T	$[-10, 30] \cdot deg2SI$	<i>inherited</i>	angle of attack
AOTSensorBias-T	electricPotentialDifference-T	$[BIAS_{\alpha}^-, BIAS_{\alpha}^+]$	<i>inherited</i>	angle of attack sensor output
AOTSensorGain-T	angularSensorGain-T	$[S_{\alpha}^-, S_{\alpha}^+]$	<i>inherited</i>	gain of angle of attack sensor
AOTSensorOutput-T	electricPotentialDifference-T	$[OR_{\alpha}^-, OR_{\alpha}^+]$	<i>inherited</i>	angle of attack sensor output
CPGain-T	float	$(-\infty, +\infty)$	$V \cdot rad^{-1}$	gain of knob controls from control panel
CPOutput-T	electricPotentialDifference-T	$[CP^-, CP^+]$	<i>inherited</i>	output of controls from control panel
DACinput-T	integer	$[0, 2^{DAC_{nb}})$	<i>inherited</i>	input to D/A card
DACoutput-T	electricPotentialDifference-T	$[DAC_{V-}, DAC_{V+}]$	<i>inherited</i>	output from D/A card
PCUInput-T	electricPotentialDifference-T	$[PCU^-, PCU^+]$	<i>inherited</i>	power control unit input
TCWeight-T	float-T	$(-\infty, +\infty)$	<i>notspecified</i>	trim condition weight
TSensorBias-T	electricPotentialDifference-T	$[BIAS_T^-, BIAS_T^+]$	<i>inherited</i>	temperature sensor output bias
TSensorGain-T	temperatureSensorGain-T	$[S_T^-, S_T^+]$	<i>inherited</i>	temperature sensor gain
TSensorOutput-T	electricPotentialDifference-T	$[OR_T^-, OR_T^+]$	<i>inherited</i>	temperature sensor output
accelerationSensorGain-T	float	$(-\infty, +\infty)$	$V \cdot s \cdot m^{-1}$	acceleration sensor gain
acceleration-T	float	$(-\infty, +\infty)$	$m \cdot s^{-1}$	linear acceleration
accelerometerBias-T	electricPotentialDifference-T	$[BIAS_A^-, BIAS_A^+]$	<i>inherited</i>	accelerometer output bias
accelerometerGain-T	accelerationSensorGain-T	$[S_A^-, S_A^+]$	<i>inherited</i>	accelerometer gain
accelerometerOutput-T	electricPotentialDifference-T	$[OR_A^-, OR_A^+]$	<i>inherited</i>	accelerometer output
aileronDeflection-T	angle-T	$[\delta_a^-, \delta_a^+]$	<i>inherited</i>	aileron deflection
airspeed-T	velocity	$[35, 65]$	<i>inherited</i>	air velocity
altitude-T	length-T	$[2000, 10000]$	$foot2SI$ <i>inherited</i>	barometric altitude
angle-T	float	$(-\infty, +\infty)$	<i>rad</i>	angle

Table C.10: Data-types (continued)

Type ID	Base type	Range	SI symbol	Description
angularAcceleration-T	float	$(-\infty, +\infty)$	$rad \cdot s^{-2}$	angular acceleration
angularSensorGain-T	float	$(-\infty, +\infty)$	$V \cdot rad^{-1}$	angular sensor gain
angularVelocitySensorGain-T	float	$(-\infty, +\infty)$	$V \cdot s \cdot rad^{-1}$	angular velocity sensor gain
angularVelocity-T	float	$(-\infty, +\infty)$	$rad \cdot s^{-1}$	angular velocity
area-T	float	$(-\infty, +\infty)$	m^2	area
attitudeSensorBias-T	electricPotentialDifference-T	$[BIAS_{\phi/\theta}^-, BIAS_{\phi/\theta}^+]$	<i>inherited</i>	attitude sensor output bias
attitudeSensorGain-T	angularSensorGain-T	$[S_{\phi/\theta}^-, S_{\phi/\theta}^+]$	<i>inherited</i>	attitude sensor gain
attitudeSensorOutput-T	electricPotentialDifference-T	$[OR_{\phi/\theta}^-, OR_{\phi/\theta}^+]$	<i>inherited</i>	attitude sensor output
bankReference-T	angle-T	$[\phi^-, \phi^+]$	<i>inherited</i>	bank reference control
controlDerivative-T	float	$(-\infty, +\infty)$	<i>notspecified</i>	control derivative
density-T	float	$[0, +\infty)$	$kg \cdot m^{-3}$	mass density
electricPotentialDifference-T	float	$(-\infty, +\infty)$	V	electric potential difference
elevatorDeflection-T	angle-T	$[\delta_e^-, \delta_e^+]$	<i>inherited</i>	elevator deflection
engineSpeed-T	revolutionPerMinute-T	$[n^-, n^+]$	<i>inherited</i>	engine RPM
flapDeflection-T	angle-T	$[\delta_f^-, \delta_f^+]$	<i>inherited</i>	flap deflection
force-T	float	$(-\infty, +\infty)$	N	force
frequency-T	float	$[0, +\infty)$	s^{-1}	force
gasConstant-T	float	287.05	$J \cdot K \cdot kg^{-1}$	gas constant
gyroBias-T	electricPotentialDifference-T	$[BIAS_g^-, BIAS_g^+]$	<i>inherited</i>	rate gyro output bias
gyroGain-T	angularVelocitySensorGain-T	$[S_g^-, S_g^+]$	<i>inherited</i>	rate gyro gain
gyroOutput-T	electricPotentialDifference-T	$[OR_g^-, OR_g^+]$	<i>inherited</i>	rate gyro output
headingReference-T	angle-T	$[\psi_r^-, \psi_r^+]$	<i>inherited</i>	heading reference control
headingSensorBias-T	electricPotentialDifference-T	$[BIAS_{\psi}^-, BIAS_{\psi}^+]$	<i>inherited</i>	heading sensor output bias
headingSensorGain-T	angularSensorGain-T	$[S_{\psi}^-, S_{\psi}^+]$	<i>inherited</i>	heading sensor gain
headingSensorOutput-T	electricPotentialDifference-T	$[OR_{\psi}^-, OR_{\psi}^+]$	<i>inherited</i>	heading sensor output
inertiaParam-T	float	$(-\infty, +\infty)$	<i>notspecified</i>	inertia parameter
length-T	float	$(-\infty, +\infty)$	m	length
moment-T	float	$(-\infty, +\infty)$	$N \cdot m$	angular moment

Table C.10: Data-types (continued)

Type ID	Base type	Range	SI symbol	Description
momentOfInertia-T	float	$[0, \infty)$	$kg \cdot m^2$	moment of inertia
natural-T	integer	$[0, +\infty)$	<i>notspecified</i>	natural number
percentage-T	float	$[0, 1]$	<i>notspecified</i>	percentage
pitchReference-T	angle-T	$[\theta^-, \theta^+]$	<i>inherited</i>	pitch reference control
pressureSensorGain-T	float	$(-\infty, +\infty)$	$V \cdot N^{-1} \cdot m^2$	pressure sensor gain
power-T	float	$(-\infty, +\infty)$	$J \cdot s^{-1}$	pressure
pressure-T	float	$(-\infty, +\infty)$	$N \cdot m^{-2}$	pressure
productOfInertia-T	float	$[0, \infty)$	$kg \cdot m^2$	product of inertia
psSensorBias-T	electricPotentialDifference-T	$[BIAS_{p_s}^-, BIAS_{p_s}^+]$	<i>inherited</i>	static pressure sensor output bias
psSensorGain-T	pressureSensorGain-T	$[S_{p_s}^-, S_{p_s}^+]$	<i>inherited</i>	static pressure sensor gain
psSensorOutput-T	electricPotentialDifference-T	$[OR_{p_s}^-, OR_{p_s}^+]$	<i>inherited</i>	static pressure sensor output
qdynSensorBias-T	electricPotentialDifference-T	$[BIAS_{q_{dyn}}^-, BIAS_{q_{dyn}}^+]$	<i>inherited</i>	dynamic pressure sensor output bias
qdynSensorGain-T	pressureSensorGain-T	$[S_{q_{dyn}}^-, S_{q_{dyn}}^+]$	<i>inherited</i>	dynamic pressure sensor gain
qdynSensorOutput-T	electricPotentialDifference-T	$[OR_{q_{dyn}}^-, OR_{q_{dyn}}^+]$	<i>inherited</i>	dynamic pressure sensor output
randomTurbulencePSD-T	float	$[0, +\infty)$	<i>notspecified</i>	random turbulence power spectral density
revolutionPerMinute-T	angularVelocity-T	$[0, +\infty)$	<i>inherited</i>	revolution per minute
rudderDeflection-T	angle-T	$[\delta_r^-, \delta_r^+]$	<i>inherited</i>	rudder deflection
sensorOutputPSD-T	float	$[0, +\infty)$	$V^2 \cdot Hz^{-1}$	power spectral density of sensor output
stabilityDerivative-T	float	$(-\infty, +\infty)$	<i>notspecified</i>	stability derivative
switch-T	boolean	$\{ON, OFF\}$	<i>notspecified</i>	switch control
temperatureGradient-T	float	$(-\infty, +\infty)$	$K \cdot m^{-1}$	temperature gradient in troposphere
temperatureSensorGain-T	float	$(-\infty, +\infty)$	$V \cdot K^{-1}$	temperature sensor gain

Table C.10: Data-types (continued)

Type ID	Base type	Range	SI symbol	Description
temperature-T	float	$(-\infty, +\infty)$	K	temperature
time-T	float	$(-\infty, +\infty)$	s	time
velocity-T	float	$(-\infty, +\infty)$	$m \cdot s^{-1}$	linear velocity
warningLight-T	boolean	$\{ON, OFF\}$	<i>notspecified</i>	warning light
warningLightInput-T	electricPotentialDifference-T	$[WL^-, WL+]$	<i>inherited</i>	input to warning light